# Aspdnsf Commander R3

User Manual Revision 1.01 (BETA)
Simrun Technologies
www.simrun.com

Copyright © 2003 - 2012 Simrun Technologies L.L.C
7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

# About Aspdnsf Commander R3

Thank you for choosing Aspdnsf Commander R3, we hope the latest version of the commander software will help you manage your online store quicker and more efficient than ever before. With R3 we have decided to create a text based manual for you, as well as a video manual, we are releasing the text based manual first with a video manual (Similar to R2) to be released shortly after, please use this manual as a guide to understand how Aspdnsf Commander R3 works in the meantime.

Please note, this manual in its current form is considered a beta manual, meaning some information, spelling, and grammatical errors may exist; we are working on fixing these issues for you and should have a final version released shortly. If you have questions regarding something that is not explained correctly, please feel free to send us an email at info@simrun.com and we'll be glad to help you (note assistance is available to registered customers).

Be sure to visit our site for any updates and information regarding Aspdnsf Commander R3:
http://www.simrun.com/AspDotNetStoreFront/AspdnsfCommander.aspx

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

# Table of Contents

CHAPTER 1: REQUIREMENTS & INSTALATION 5

- ASPDOTNETSTOREFRONT MINIMUM REQUIREMENTS
- INSTALLING WSI FILES
- UPDATING WEB.CONFIG
- ASPDNSF COMMANDER R3 MINIMUM REQUIREMENTS
- INSTALLING ASPDNSF COMMANDER R3
- TESTING CONNECTION & CONFIGURATION

CHAPTER 2: EXPORTS WITH ASPDNSF COMMANDER R3 11

- PERFORMING PRODUCT EXPORTS
- PERFORMING KIT EXPORTS
- PERFORMING ENTITY EXPORTS
- SQL EXPORTS
- CUSTOMER & ORDER EXPORTS
- SCHEDULED EXPORTS
- PERFORMANCE TIPS & RECOMMENDATIONS

CHAPTER 3: IMPORTS, UPDATES & DELETES WITH ASPDNSF COMMANDER R3 21

- PERFORMING PRODUCT IMPORTS
- PERFORMING ENTITY IMPORTS
- PERFORMING KIT IMPORTS
- PERFORMING UPDATES
- UPDATE PARSING
- PERFORMING DELETES
- PROCESSING YOUR IMPORT/UPDATE/DELETE FILES

CHAPTER 4: SCHEDULER 30

- USING SCHEDULER TO SETUP ROUTINE TASKS
- STATUS WINDOW & LOG FILES
- TIPS & RECOMMENDATIONS

CHAPTER 5: IMAGE HANDLING 33

- IMAGE UPLOADING VIA CSV FILES
- IMAGE UPLOADING VIA THE IMAGES TAB

Copyright © 2003 - 2013 Simrun Technologies L.L.C. 
7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

CHAPTER 6: DATA & DATA TYPES 36

- PRODUCT/VARIANT CSV DATA AND COLUMN NAMES
- ENTITY (CATEGORY & SECTION) CSV DATA AND COLUMN NAMES
- ENTITY (MANUFACTURER) CSV DATA AND COLUMN NAMES
- KIT CSV DATA & COLUMN NAMES

CHAPTER 7: ASPDSNF COMMANDER R3 LICENSE 50

- REQUESTING A NEW KEY
- UPGRADE YOUR OLD R2 KEY

Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

# Chapter 1: REQUIREMENTS & INSTALATION

## ASPDOTNETSTOREFRONT WSI IPX MINIMUM REQUIREMENTS

Aspdnsf Commander R3 requires the WSI Interface which was introduced for AspDotNetStorefront in version 7.23, hence the minimum requirements to run Aspdnsf Commander R3 is the ML 7.23 version of AspDotNetStorefront. Note some features may not be available in versions below the Multi-Store version of 9.X MSX and as such the minimum version officially supported by Commander is 9.X MSX starting with the R3 generation of Commander.

While Commander R3 will work on any version above 7.23 product support from Simrun Technologies is only offered on 9.X MSX and higher due to changes in the WSI system and on-going security patches released by Aspdotnetstorefront. If you are running a version of Aspdotnetstorefront older than 9.X MSX, please consider upgrading as the newer 9.X MSX provides new functionality and greater stability over previous releases and ensures that the latest third party integrations work properly with your cart.

Aside from the minimum version of AspDotNetStorefront you will need the following to install the WSI component to your site:

- FTP Root Access to the directory that hosts your site (contact your web host for more details)
- A Super User Admin and password to your site
- Access to the web.config file and the ability to edit the file to enable the WSI component
- Access to the IPX and WSI files for your particular version so that you can install the files needed to run the WSI component.
- Ensure that the WSE 3.0 Add-on is installed on your Server (Windows 200 and Windows 2003 ONLY, windows 2008 already has this installed)
- OPTIONAL: Access to your IIS Server, or a way to restart you website after you have made the changes. If this is not available then you'll need to contact your host to restart your website after you have installed the WSI interface.

Once you have all the above you are ready to install the WSI component so Commander R3 can connect to your website.

## INSTALLING WSI IPX FILES

The first step in installing the WSI interface is to install the WSI files in your Aspdotnetstorefront site so that you have an interface from which the commander can "Talk" to your site. This is fairly straight forward process and only requires you copy files that were included in your installation zip/folder over

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

to your production site. Begin by looking in your install directory (the place you extracted your aspdotnetstorfront files) and look for a folder called WSI from this folder you'll need to do the following:

1. Copy the ASPDNSFUserNameTokenManager.cs and ipx.cs files into the {root}/App_Code folder on your site.

2. Copy the ipx.asmx and ipx.xml files from the WSI folder to the root of your site.

At this point you can rename the IPX.asmx file to something else If you want (for security purposes) or you can leave it as is, do not rename the extension, this should remain .asmx.

Once you have done this you have installed the WSI Interface files, now all that remains is to update the web.config and ensure that the WSI URL is working properly.

## UPDATING WEB.CONFIG

The WSI Interface requires that you modify the web.config to enable the WSI component, this step is a bit tricky but is required to enable the WSI Interface. To begin download the web.config from your root folder via ftp and save a copy of it as a backup. Now open your web.config with visual studio if you have it, or using NOTEPAD.

Look for the commented lines related to the WSI they should say something along the lines:

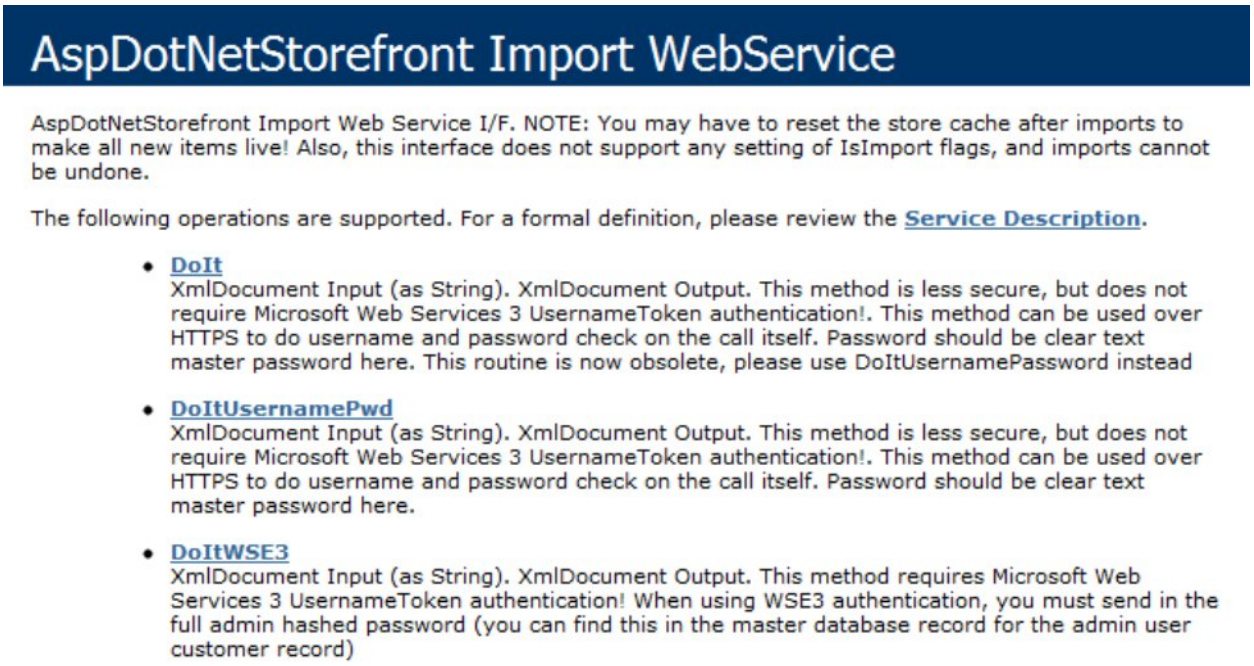
```
<!-- WSI Web Service Interface  -->
<!-- There are four separate sections within this file that need to be un commented. This
is section 1 of 4. -->
<!-- WSI Web Service Interface (Section 1 of 4) Uncomment the following line to enable
WSI Web Service Interface-->

<!--
<section name="microsoft.web.services3" type="Microsoft.Web.Ser
vices3.Configuration.WebServicesConfiguration,
Microsoft.Web.Services3, Version=3.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
-->
```

You'll need to remove the "<!—"and the "-->" to uncomment the line the, area that needs to be uncommented has been bolded and enlarged for you. There are 4 areas where you have to uncomment WSI Interface items to enable the WSI Interface. PLEASE NOTE: Not all installs will work with all 4 areas of the WSI commented out. You may need to uncomment 1 of 4, 2 of 4 and 3 of 4 but leave 4 of 4 commented on some installs to get the Commander R3 to connect properly. If this still does not work you try commenting step 3 of 4. If that still does not work, contact Simrun Support (For registered and active support holding customers only).

Once you have done this, ftp the updated web.config file back to your root folder and go to your website and navigate to your IPX URL (WSI Interface) e.g. www.yourdomain.com/ipx.asmx (Note if you

Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

renamed your .asmx page than you'll need to go to that file). Upon arriving to the page you should see something like this:


AspDotNetStorefront Import WebService
AspDotNetStorefront Import Web Service I/F. NOTE: You may have to reset the store cache after imports to make all new items live! Also, this interface does not support any setting of IsImport flags, and imports cannot be undone.
The following operations are supported. For a formal definition, please review the Service Description.
DoIt
XmlDocument Input (as String). XmlDocument Output. This method is less secure, but does not require Microsoft Web Services 3 UsernameToken authentication!. This method can be used over HTTPS to do username and password check on the call itself. Password should be clear text master password here. This routine is now obsolete, please use DoItUsernamePassword instead
DoItUsernamePwd
XmlDocument Input (as String). XmlDocument Output. This method is less secure, but does not require Microsoft Web Services 3 UsernameToken authentication!. This method can be used over HTTPS to do username and password check on the call itself. Password should be clear text master password here.
DoItWSE3
XmlDocument Input (as String). XmlDocument Output. This method requires Microsoft Web Services 3 UsernameToken authentication! When using WSE3 authentication, you must send in the full admin hashed password (you can find this in the master database record for the admin user customer record)

If you see this page, then your WSI component is installed (the files at least) NOTE: Even though the page displays it may not mean your web.config is setup correctly as you may still not be able to connect, we'll verify this when we try to have Aspdnsf Commander R3 connect to your site's WSI interface.

Once you have completed this step your website now has a functioning interface from which Commander R3 can talk to, now the fun begins as we can install Aspdnsf Commander R3 and make you site managing a whole lot easier.

## ASPDNSF COMMANDER R3 MINIMUM REQUIREMENTS

Aspdnsf Commander R3 now in its 3rd iteration is now more compatible than ever, but it still remains a windows only program. Commander R3 requires these minimum requirements:

- Windows XP w/Sp3 or higher (windows 7 professional recommended)
- Asp.net 2.0 and higher
- 2.5 Ghz CPU or higher (Any Dual Core 2.0 ghz or higher is recommended)
- 2 GB ram or higher (4GB ram or higher recommended
- A broadband internet connection (512 Kbs) or higher

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

- 200 Mb install space (more will be needed to store log files and image resizing, 2-4 Gigabytes of free space recommended)

If using professional version of any windows OS you will need administrator rights and the ability to run programs as an Administrator (i.e. right click and select run as Administrator on windows vista and windows 7).

## TESTING CONNECTION & CONFIGURATION

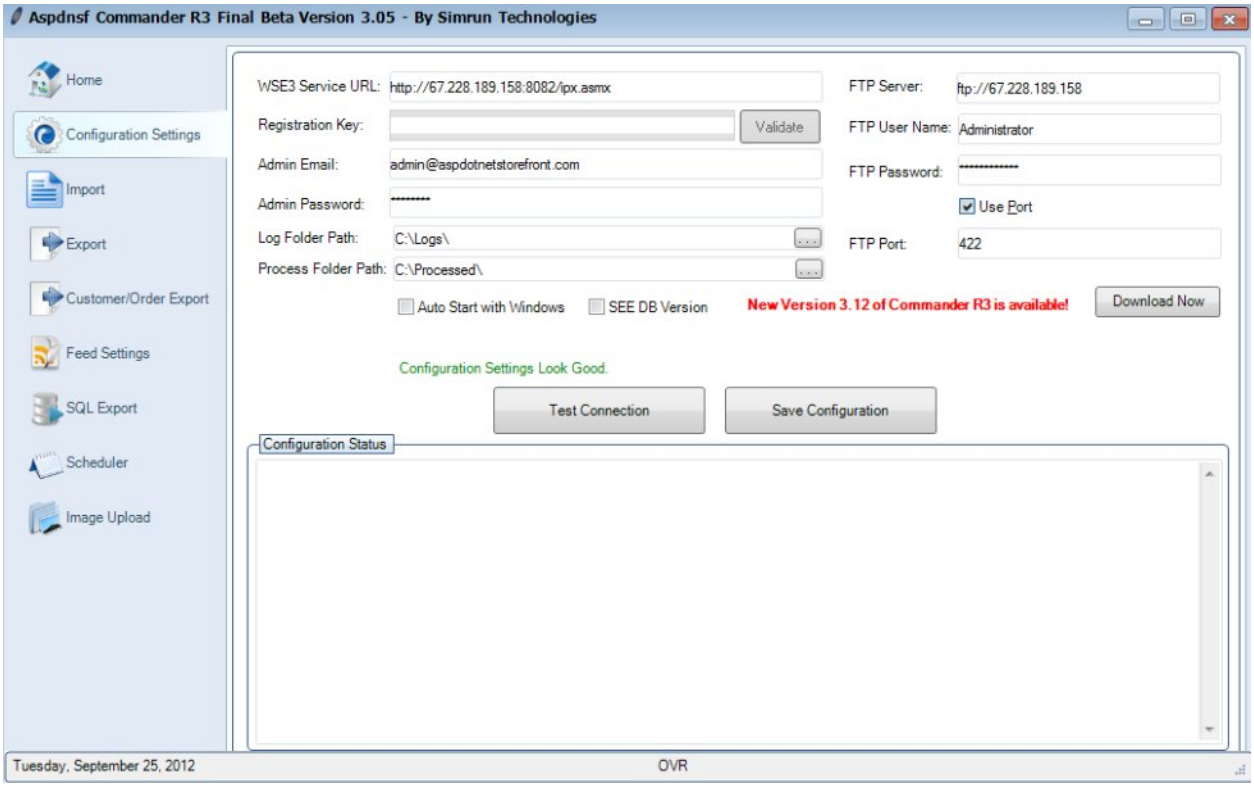
To download Commander visit the Simrun Website: http://www.simrun.com/AspDotNetStoreFront/AspdnsfCommander.aspx and download the latest release.

The installer is fairly straight forward, chose your destination, decide who has access to the program and you will have the program installed with your selected options. You'll need to agree to the legal disclaimer and waiver of liability before you can proceed, make sure you understand this; for 99.999% of the folks that will be using Commander R3, this is known as "The Legal Mumbo Jumbo", and is the same as the countless disclaimers on various software, but never the less it's there and must be agreed upon before you can install it on to your machine.

NOTE: Commander R3 may install new stored procedures and database fields on to your sites database. This is needed to ensure proper functionality and does not hamper the stock AspDotNetStorefront's functions in anyway. If you need details on what changes are made to the database please contact Simrun Support at info@simrun.com.

After Aspdnsf Commander R3 has been installed, please start the program, you will see a splash screen and then a configuration screen:

Copyright © 2003 - 2013 Simrun Technologies L.L.C.




Here you'll need to enter the following information:

WSE3 Service URL: This is your IPX url (NOT YOUR WEBSITE ADMIN) e.g. http://www.yourdomain.com/ipx.asmx make sure you include the HTTP:// prefix, without it Commander won't connect

Registration Key: Optional, If you have a key, please enter the key here, note: in order to validate the key you'll still need to have valid super user information and your WSE 3 url information as Commander needs to connect to your site to validate the key.

RETAIL VERSION: The retail version of commander R3 will not ask you for a registration key, instead you will provided with a .licx license file that you will need to place into the root folder where you installed commander. After you place the file in the root, restart commander and your license information will be displayed inside the ABOUT tab.

Admin Email: Your SUPER ADMIN login email address (this is the same login that you use to log into your site's admin)

Admin Password: Your SUPER ADMIN password (the same password you use to log into your site's admin)

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

Log Folder Path: This is the directory where Commander will store its logs of activities, you can change this to any directory that you have write access to. NOTE this folder can get fairly big so make sure you enough space, you can also delete old log files that you no longer need.

Processed Folder Path: This is the folder where commander will move processed files such as completed import files, or resized images, NOTE this folder can get fairly big so make sure you enough space, you can also delete old processed files that you no longer need.

FTP Settings: These should be fairly self-explanatory, enter in your FTP settings so that commander can download and upload images to you site as well as download them.

Once you have entered in your settings, click the Test Connection Button, Commander will let you know if your WSI and FTP settings are working, at a minimum you need to have WSI settings working before the program "unlocks itself" but its recommended you get the FTP working as well as R3 uses this to upload files to and from the website.

If you get an error saying Connection Failed then you need to ensure that your WSI is installed properly (see Installing WSI in this manual), other reasons could be:

Can't Connect/Failed:
If you are sure that the WSI is installed properly; Make sure you are using a SUEPR ADMIN LOGIN as regular Admin Logins do not work. Also verify your password hasn't expired by trying to log into your website admin. If you have followed the directions and STILL CAN NOT CONNECT, Please Leave Step 4 in the webconfig COMMENTED, this is because on some version of ML 8 and MSX v9 the commander has issues with the tokenmanager.cs, this issue is further explained here:
http://manual.aspdotnetstorefront.com/p-1213-wsi-sample-test-program-not-working-with-downloadable-demo.aspx

WSI Page Not displaying/Working:
Make sure you have WS3 Web Services Enhancements installed on your machine and you also have installed the WSI component properly. Check the manual for more help:
http://manual.aspdotnetstorefront.com/wsi/ and view the installing WSI in this manual.

If your connection is successful, click the Save Configuration button, upon doing this you have successfully installed Aspdnsf Commander R3 and now the rest of the program is available for you to use.

Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

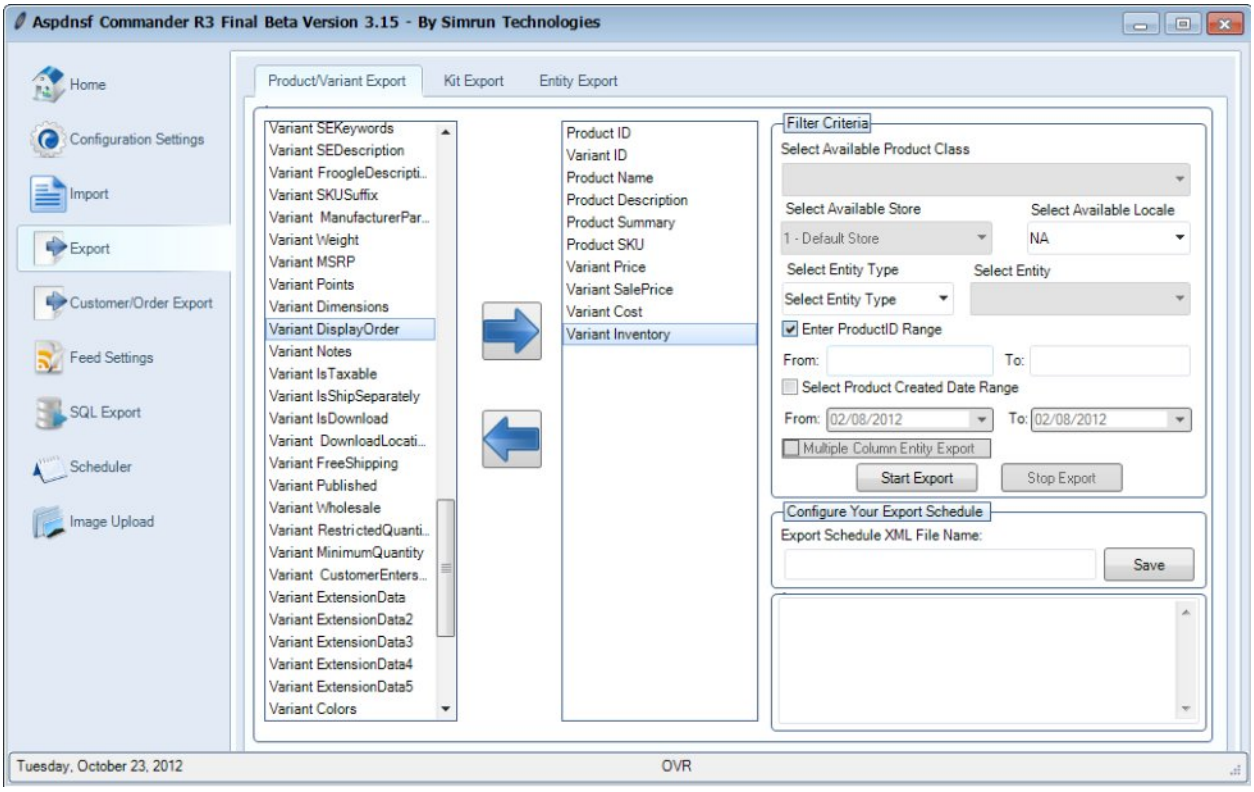
# Chapter 2: EXPORTS WITH ASPDNSF COMMANDER R3

## PERFORMING PRODUCT EXPORTS

Contrary to what many users may want to do first (as in import data), we recommend you start using Aspdnsf Commander by first performing some basic exports to familiarize yourself with the data format and field names in Aspdnsf Commander R3. This will teach you how Commander handles data and will give you a very good grasp of what you need to do to perform imports and updates.

In Aspdnsf Commander, Exports are almost identical to update and insert files with the exception that they have different minimum requirements and certain fields are not needed (such as the Action field which is explained later in the Imports / updates section) for the purposes of this manual, we'll be doing exports first and then using the export file as a basis for future import / update files.

To perform your first export, click on the export slider tab:




Once you click on this tab you'll see a list of Available fields that you can select to export into a CSV file. By default when you first click on the export tab you are displayed the Product/Variant Export options. This tab with in the export options is what you'll be using majority of the time as it allows you to export

product and variant information and all information pertaining to those items (such as name, description, prices as well as supplement information such as categories, sections, and manufacturers they are mapped to).

You'll also notice that 2 fields are pre-selected such as the Product ID and Variant ID. These fields are extremely important as the serve as the primary keys for products and variants, hence in order to reference unique records commander automatically selects the main IDs for both products and variants.

Once you have selected some fields let's look at the column to the right, the Filter Criteria; these options allow you to fine tune your export so that you can get the data that you want and leave out the records that you don't' need.

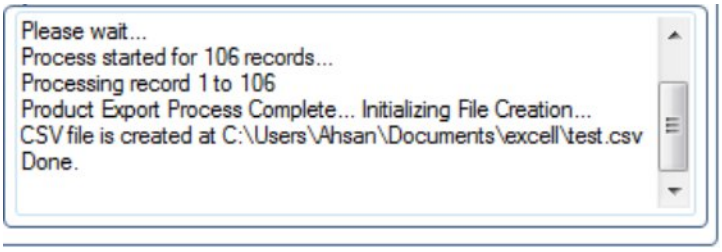
The first option is the Product Class option, this only applies to S.E.E. Simrun developed Sites and will now applied to stock sites; in a nut shell the product class allows to export a specific group of items form your store (See the S.E.E. Only Features Section for more information).

If you have a Multi-Store Site with multiple domains, you can select the domain that you want to export items from, e.g. if you have 2 or 3 sites, but only need to export sites from store 2, simply select that store ID/store name from the Select Available Store drop down and Aspdnsf Commander R3 will restrict its results from that particular store.

After making your selection you can than tell Aspdnsf Commander R3 to export items that belong to a certain category, section, or manufacturer. To do this simply select the entity type from the Select Entity Type drop down and then select the name of the entity you want to export form the Select Entity drop down.

Finally if you need to restrict your export by a certain product ID range or product created on date range, you can select the starting and ending values in the Enter Product ID Range and Select Product Created Dated Range options respectively.

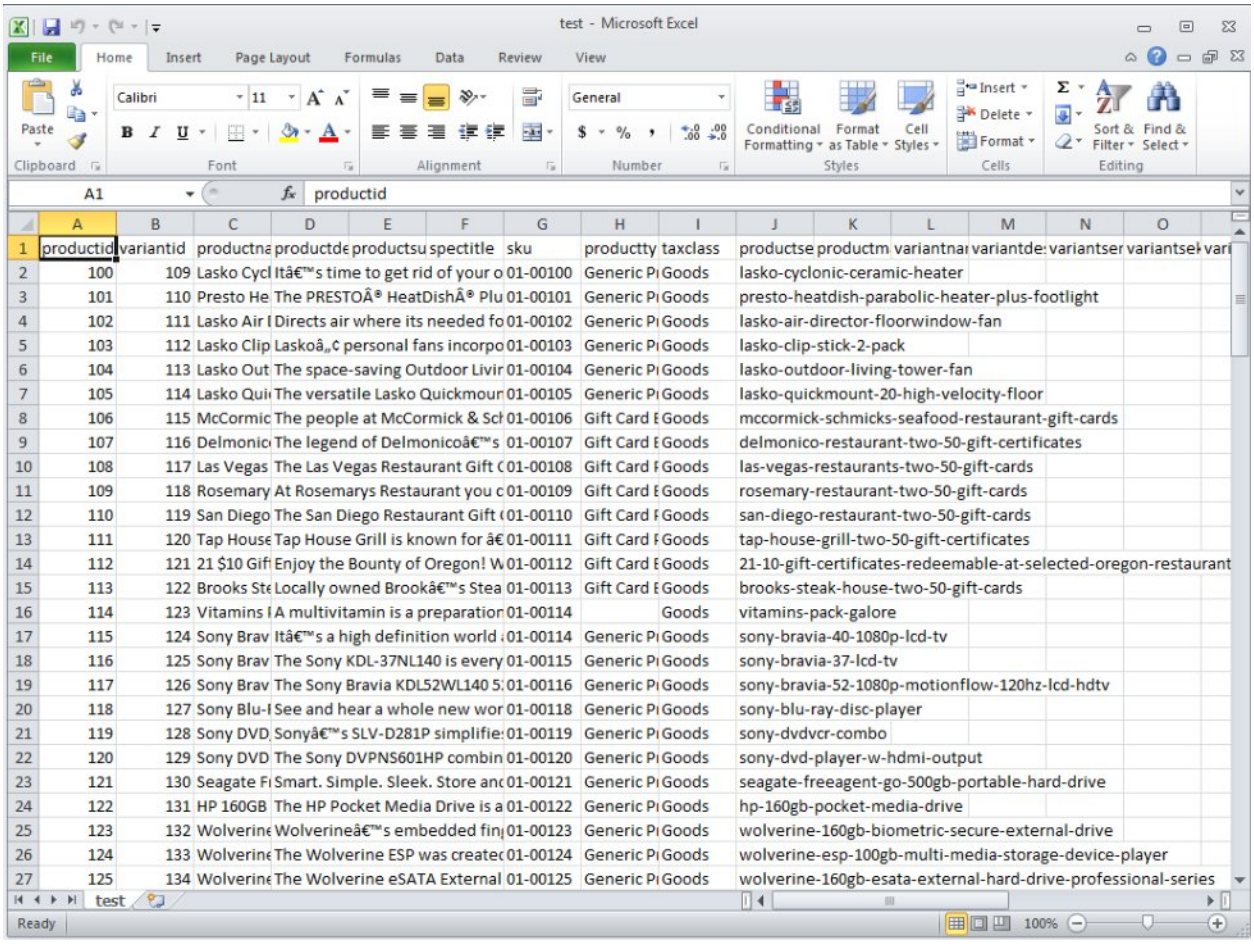
Now you are ready to perform your export NOTE: you may notice an option called Multiple Column Entity Export for more information on this please see allowed data types and data in chapter 5. Click on the start export and Aspdnsf Commander R3 will begin the process of exporting the data from your site, a status window will tell you the status regarding how many matches it found for the criteria you selected as well as how many records it has downloaded so far.


Please wait...
Process started for 106 records...
Processing record 1 to 106
Product Export Process Complete... Initializing File Creation...
CSV file is created at C:\Users\Ahsan\Documents\excell\test.csv
Done.

After all the records are downloaded Aspdnsf Commander will ask you where to save the file, the file will be saved as a .CSV file.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

Opening the csv file you should see something similar to this:




The data above is from the default AspDotNetStorefront products database that are included with the install package. Aspdnsf Commander organizes data at the variant level, meaning each row represents 1 variant. You'll see in the export that some rows have the same product id, however, each row has a unique variant ID, this is because products that have more than 1 variant will be listed as 2 lines on the export file; this is to help you group products and their respective variants together.

To see how different information exported try performing different exports with different data, you can also view the data types and data format table to see what type of information is exported. This is useful in case you are wondering why some fields have "0" and "1" instead of true are false (Boolean Values) as well as why some fields have multiple values inside 1 column with separated semi-colons why other fields have individual numbered fields.

Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

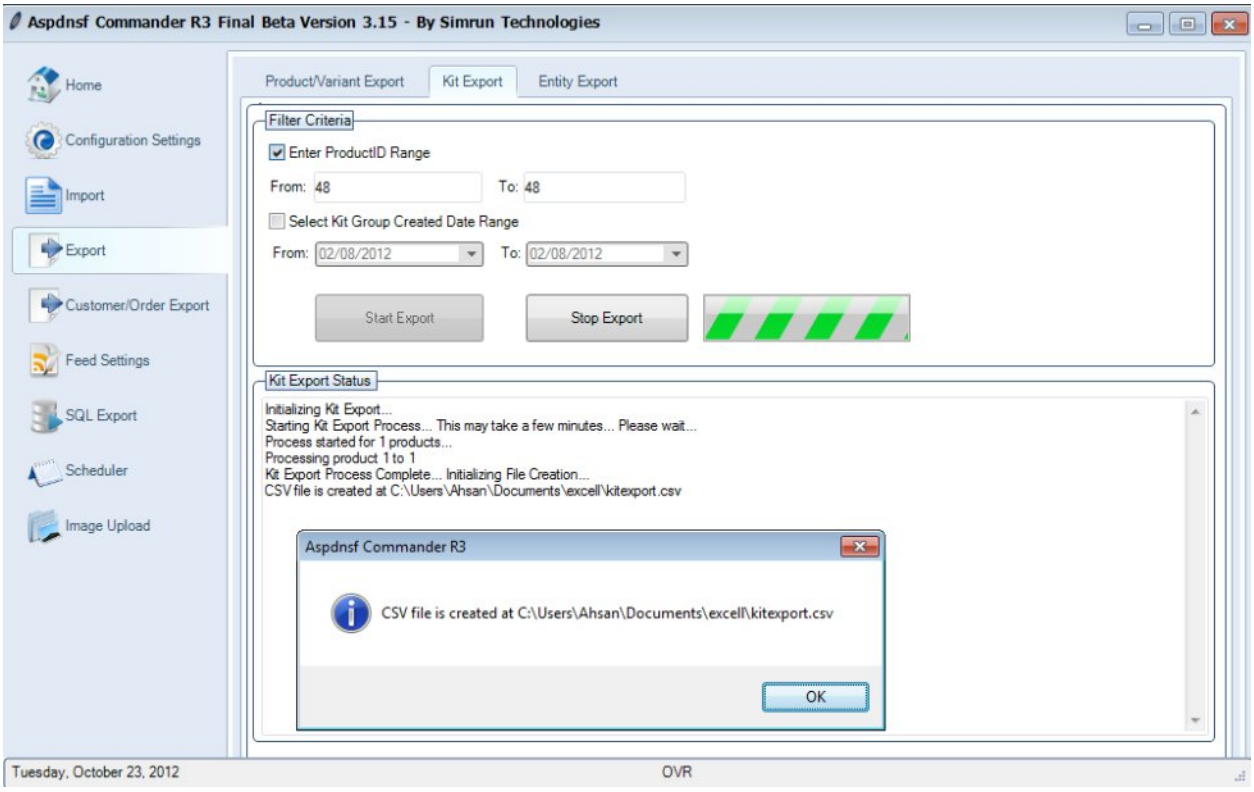
# PERFORMING KIT EXPORTS

Kits are more complex products in nature than standard variant based items on AspDotNetStorefront, and while many stores will never need to worry about kits, there are companies that rely on product kits to make the buying process intuitive for shoppers.

In Aspdnsf Commander R3 we separate kit information from general product information to make managing data easier in a .csv file environment. This is done primarily to ensure easier data entry and management.

To access kit exports click on the Kit Export tab inside the main export slide tab once there you can select the products that you would like to manage, unlike the product export Kit exports have a fixed no. of fields that are always exported (about 15).

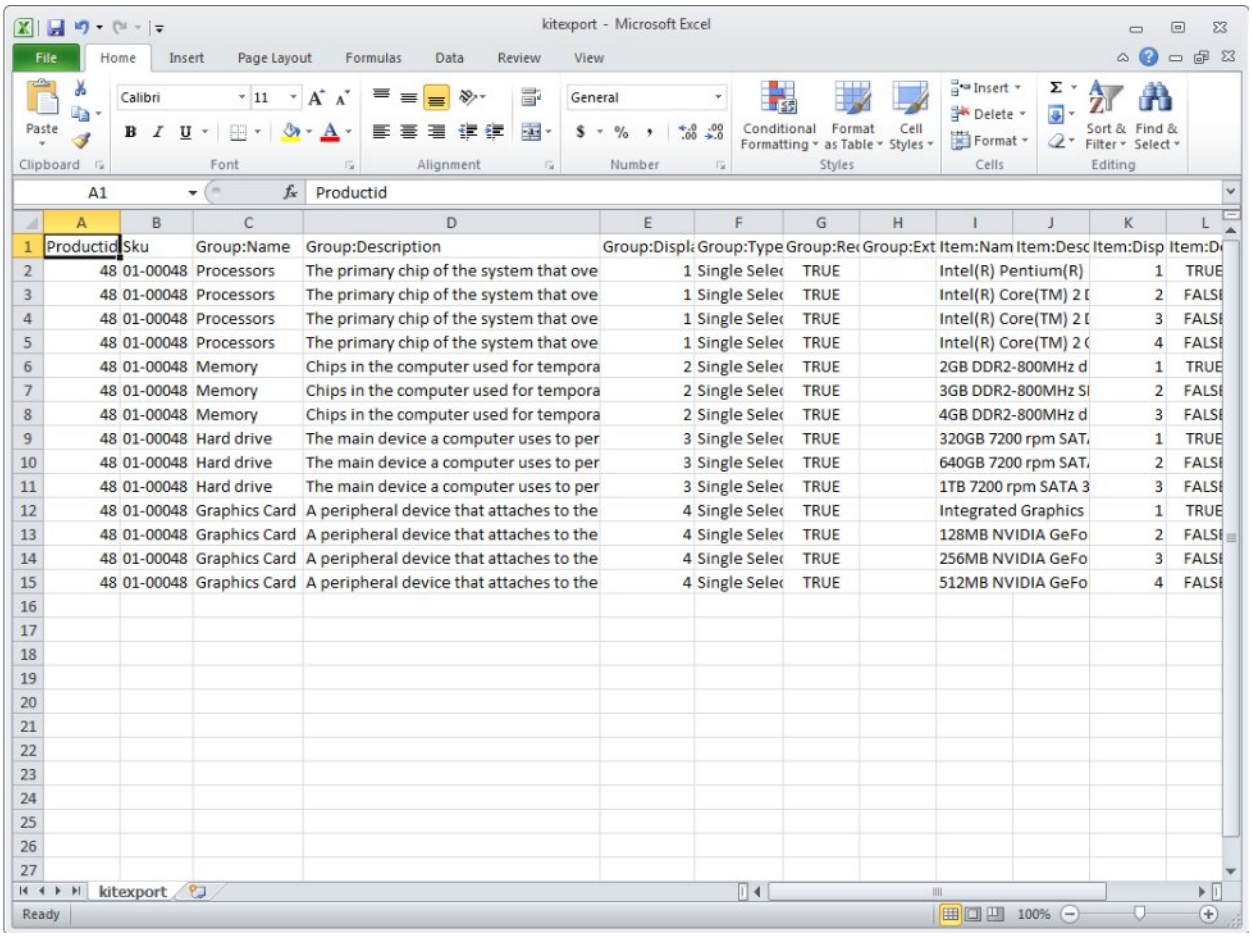
You can select the ID range or date created range for the kits, any product that is set to kit product will be exported out that matches your selection criteria. Once again, as in products Aspdnsf Commander R3 will ask you were to save this information after you click on Start Export




Copyright © 2003 - 2013 Simrun Technologies L.L.C.

7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

Once exported, you can now view your exported file



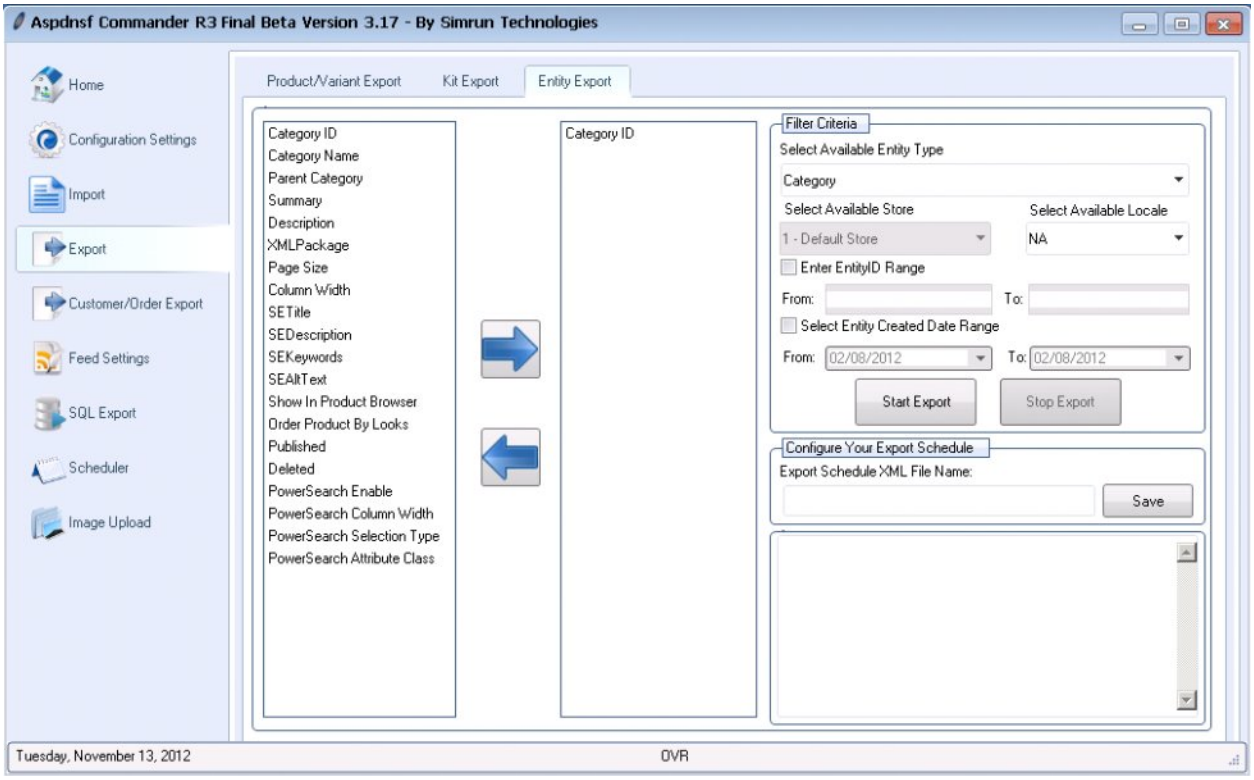

| Productid | Sku | Group:Name | Group:Description | Group:Displ | Group:Type | Group:Re | Group:Ext | Item:Nam | Item:Desc | Item:Disp | Item:D |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 48 | 01-00048 | Processors | The primary chip of the system that ove | 1 | Single Selec | TRUE | | Intel(R) Pentium(R) | | 1 | TRUE |
| 48 | 01-00048 | Processors | The primary chip of the system that ove | 1 | Single Selec | TRUE | | Intel(R) Core(TM) 2 D | | 2 | FALSE |
| 48 | 01-00048 | Processors | The primary chip of the system that ove | 1 | Single Selec | TRUE | | Intel(R) Core(TM) 2 D | | 3 | FALSE |
| 48 | 01-00048 | Processors | The primary chip of the system that ove | 1 | Single Selec | TRUE | | Intel(R) Core(TM) 2 C | | 4 | FALSE |
| 48 | 01-00048 | Memory | Chips in the computer used for tempora | 2 | Single Selec | TRUE | | 2GB DDR2-800MHz d | | 1 | TRUE |
| 48 | 01-00048 | Memory | Chips in the computer used for tempora | 2 | Single Selec | TRUE | | 3GB DDR2-800MHz SI | | 2 | FALSE |
| 48 | 01-00048 | Memory | Chips in the computer used for tempora | 2 | Single Selec | TRUE | | 4GB DDR2-800MHz d | | 3 | FALSE |
| 48 | 01-00048 | Hard drive | The main device a computer uses to per | 3 | Single Selec | TRUE | | 320GB 7200 rpm SAT/ | | 1 | TRUE |
| 48 | 01-00048 | Hard drive | The main device a computer uses to per | 3 | Single Selec | TRUE | | 640GB 7200 rpm SAT/ | | 2 | FALSE |
| 48 | 01-00048 | Hard drive | The main device a computer uses to per | 3 | Single Selec | TRUE | | 1TB 7200 rpm SATA 3 | | 3 | FALSE |
| 48 | 01-00048 | Graphics Card | A peripheral device that attaches to the | 4 | Single Selec | TRUE | | Integrated Graphics | | 1 | TRUE |
| 48 | 01-00048 | Graphics Card | A peripheral device that attaches to the | 4 | Single Selec | TRUE | | 128MB NVIDIA GeFo | | 2 | FALSE |
| 48 | 01-00048 | Graphics Card | A peripheral device that attaches to the | 4 | Single Selec | TRUE | | 256MB NVIDIA GeFo | | 3 | FALSE |
| 48 | 01-00048 | Graphics Card | A peripheral device that attaches to the | 4 | Single Selec | TRUE | | 512MB NVIDIA GeFo | | 4 | FALSE |

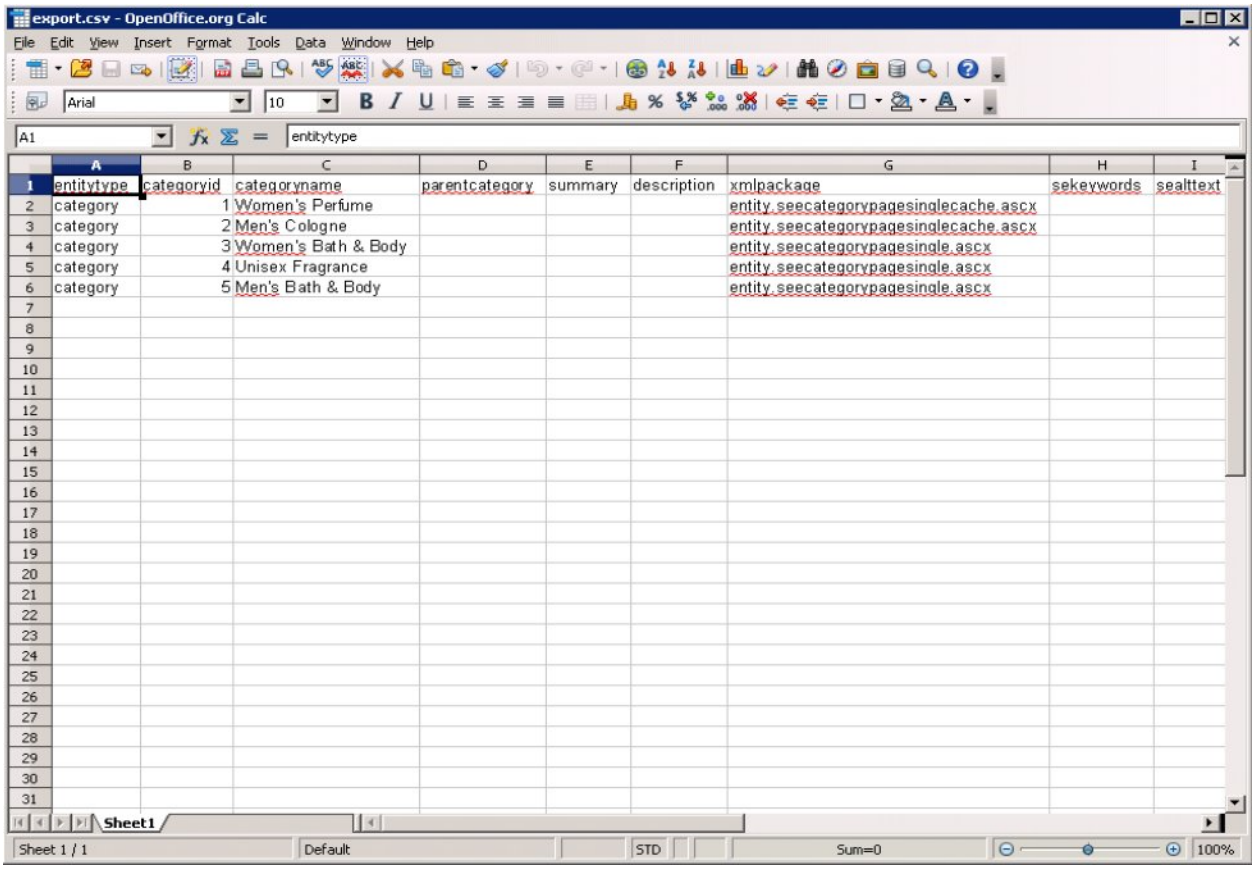
Similar to product exports, each row represents an item in the kit group; this is evident by the product ID column which remains the same for all items that belong to the particular kit. NOTE the ids for kit items and groups are not exported to keep the export simple (this is because the AspDotNetStorefront admin doesn't provide an easy way to view the group and kit Item IDS and based on user feedback many users reference kit information based on the name).

## PERFORMING ENTITY EXPORTS

Aspdnsf Allows you to export entity level information from your Aspdotnetstorefront site from 3 entity types: Categories, Sections (Departments) and Manufacturers. The entity export, similar to the product export tab allows you to select the fields you would like to export. You can also drill-down the filtering by selection a specific store, Locale, Entity ID range, or date created range.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.



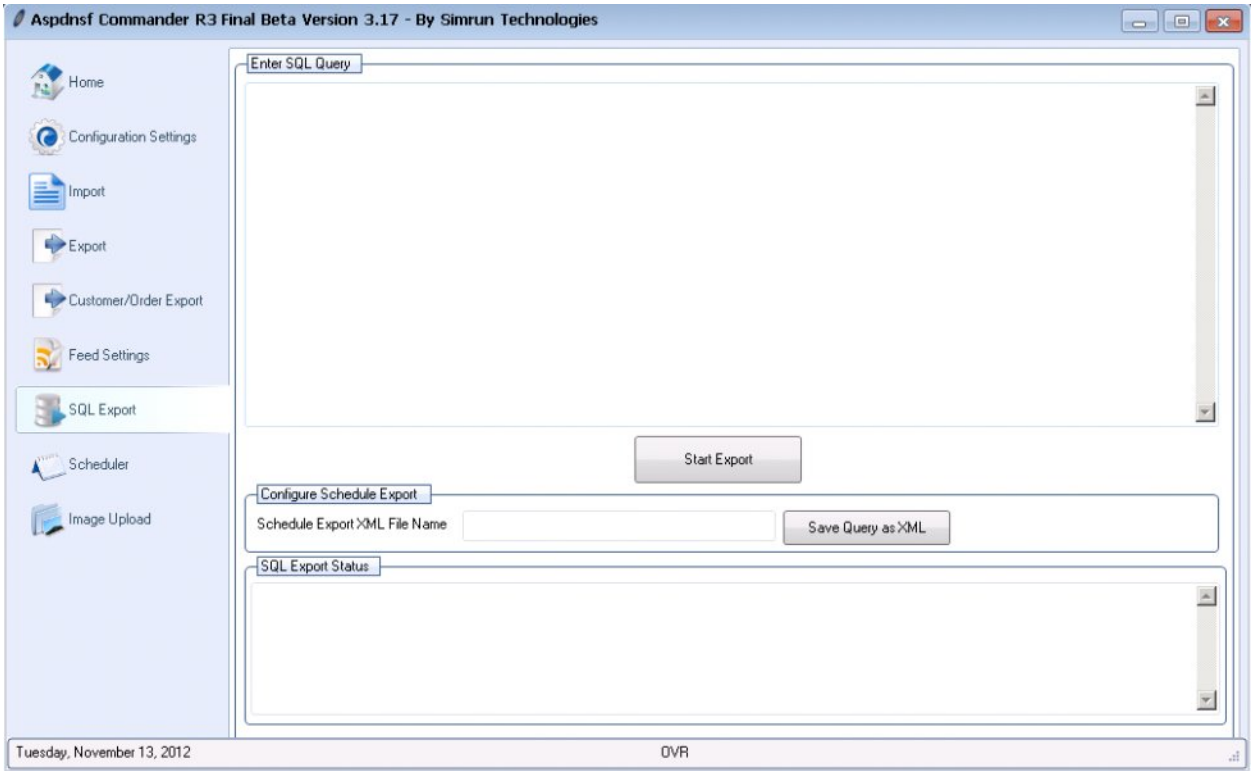




Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

Aspdnsf Commander does not support unsupported entities such as Libraries, Genres, and vectors as these are not officially supported by Aspdotnetstorefront. In the exported file, each row represents a single entity and all information pertaining to that entity is displayed in each row.

## SQL EXPORTS

While Aspdnsf Commander includes the ability to export the most commonly used information from your Aspdotnetstorefront site, there are always situations where you may need a custom query or report that is not readily available in Aspdnsf Commander. For these requirements, a SQL Export option is available that will allow you to save in a csv file the results of custom queries. NOTE: An in-depth knowledge of SQL and the Aspdotnetstorefront database structure is required before using this feature, if you need a custom sql export created, contact Simrun or other qualified AspDotNetStorefront developers that can assist you in created a sql query to run.



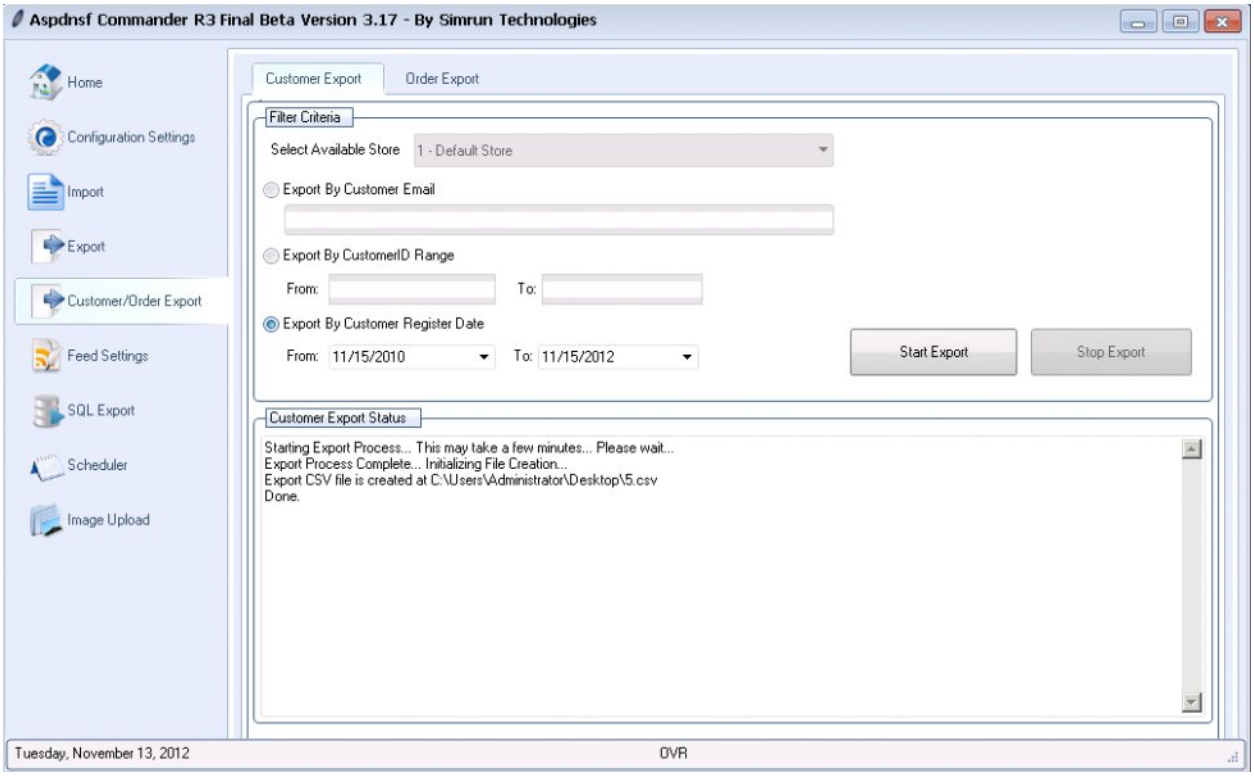

After queries are run, Aspdnsf commander will prompt you to save the .csv file which contains the information from the query that was requested.

## CUSTOMER & ORDER EXPORTS

Aspdnsf Includes basic customer and order export functionality. While these options shall be expanded in the future, current versions of Aspdnsf Commander will only a fixed no. of fields to be exported for both orders and customers.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

The customer and order export options are available in the Customer/Order Export tab, once here you can select from either sub tab to download information inside a csv file. Both Customer and Order exports create a csv file with information from the AspDotNetStorefront database. In the case of customers, all customer related fields including billing and shipping address and registration information is exported. Some basic filtering options are included such as limiting the exports based on customer email, date range and registration date, if you are using multi-store and have isolated your customers you can also select the Store ID from where you would like to perform your export.




NOTE: Customer passwords are exported in HASH values (encrypted) this cannot be changed as Aspdnsf does not store un-encrypted passwords.

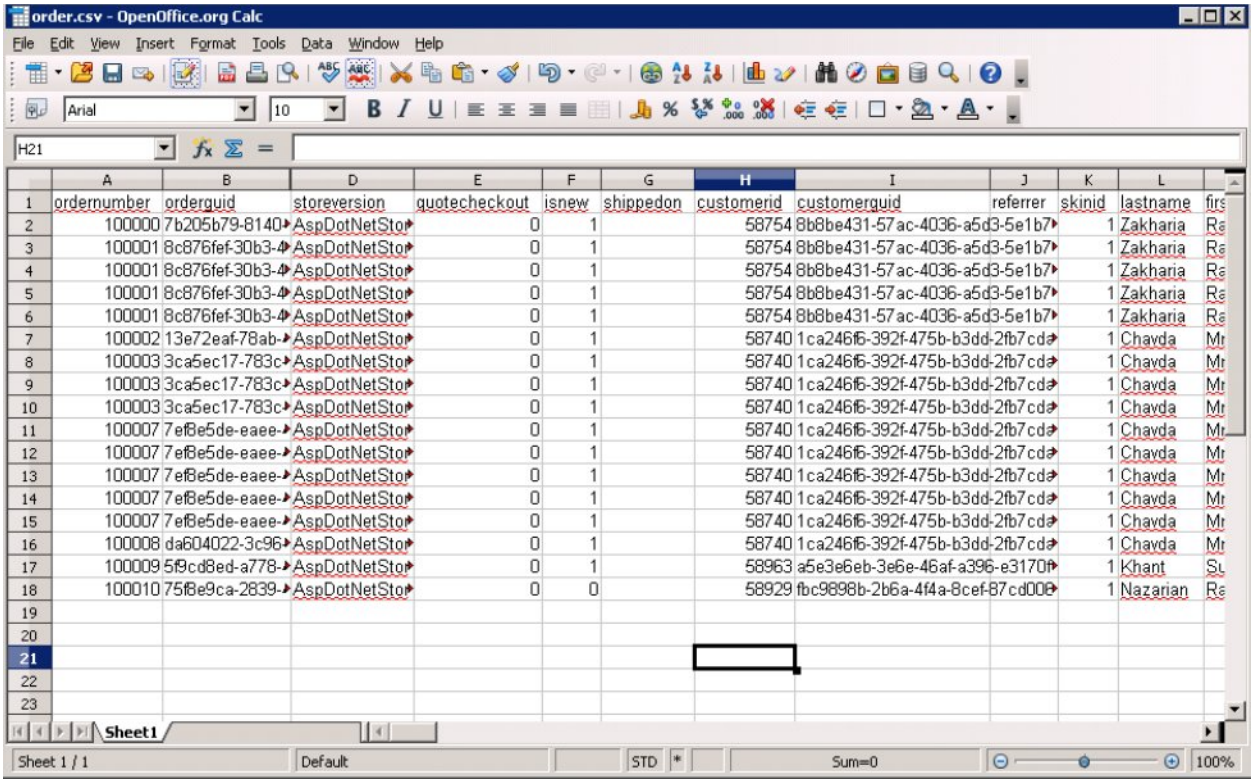
The exported file will list each customer's information per row, as in each row represents a different customer and all information pertaining to that customer is exported. NOTE: Anonymous customers are not exported; however, non-registered customer information is included in the export.

Performing Order exports is very similar to the Customer Export Screen with similar pre-filtering criteria, and file output. In future versions of Aspdnsf Commander we may add support for fine tuning order export options but for now a "RAW" dump of database information is exported in a csv file format.

The Exported orders are in a format as follows: Each row represents 1 item from an order, as in 1 order may have 5 or rows of items, this can be seen by the order number (the first column in the export file) remains the same for all the items that are a part of the order.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

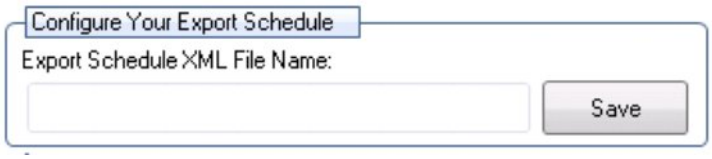
7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com




| | A | B | D | E | F | G | H | I | J | K | L | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | ordernumber | orderguid | storeversion | quotecheckout | isnew | shippedon | customerid | customerguid | referrer | skinid | lastname | firs |
| 2 | 100000 | 7b205b79-8140- | AspDotNetStor | 0 | 1 | | 58754 | 8b8be431-57ac-4036-a5d3-5e1b7 | | 1 | Zakharia | Ra |
| 3 | 100001 | 8c876fef-30b3-4 | AspDotNetStor | 0 | 1 | | 58754 | 8b8be431-57ac-4036-a5d3-5e1b7 | | 1 | Zakharia | Ra |
| 4 | 100001 | 8c876fef-30b3-4 | AspDotNetStor | 0 | 1 | | 58754 | 8b8be431-57ac-4036-a5d3-5e1b7 | | 1 | Zakharia | Ra |
| 5 | 100001 | 8c876fef-30b3-4 | AspDotNetStor | 0 | 1 | | 58754 | 8b8be431-57ac-4036-a5d3-5e1b7 | | 1 | Zakharia | Ra |
| 6 | 100001 | 8c876fef-30b3-4 | AspDotNetStor | 0 | 1 | | 58754 | 8b8be431-57ac-4036-a5d3-5e1b7 | | 1 | Zakharia | Ra |
| 7 | 100002 | 13e72eaf-78ab- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 8 | 100003 | 3ca5ec17-783c- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 9 | 100003 | 3ca5ec17-783c- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 10 | 100003 | 3ca5ec17-783c- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 11 | 100007 | 7ef8e5de-eaee- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 12 | 100007 | 7ef8e5de-eaee- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 13 | 100007 | 7ef8e5de-eaee- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 14 | 100007 | 7ef8e5de-eaee- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 15 | 100007 | 7ef8e5de-eaee- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 16 | 100008 | da604022-3c96- | AspDotNetStor | 0 | 1 | | 58740 | 1ca246f6-392f-475b-b3dd-2fb7cd | | 1 | Chavda | Mr |
| 17 | 100009 | 5f9cd8ed-a778- | AspDotNetStor | 0 | 1 | | 58963 | a5e3e6eb-3e6e-46af-a396-e3170f | | 1 | Khant | Su |
| 18 | 100010 | 75f8e9ca-2839- | AspDotNetStor | 0 | 0 | | 58929 | fbc9898b-2b6a-4f4a-8cef-87cd00 | | 1 | Nazarian | Ra |
| 19 | | | | | | | | | | | | |
| 20 | | | | | | | | | | | | |
| 21 | | | | | | | | | | | | |
| 22 | | | | | | | | | | | | |
| 23 | | | | | | | | | | | | |

## SCHEDULED EXPORTS

On product, entity and sql exports you will notice a “Export Schedule XML File Name” prompt area with a save button, this option allows you to save your export configuration and re-load it for later use in case you want to export items at a later date or run the same export again based on the criteria you have selected.




The export scheduler will allow you to save an export configuration and run it later in the Scheduler tab, for more details see chapter 4.

## PERFORMANCE TIPS & RECOMMENDATIONS

As a general rule, export only the information you need, it’s not wise to simply say “Ohh I’ll just export everything” while yes you can do this, your exports will take quite a while to complete and it’s not good practice especially on slow internet connections. We recommend you decide the data you need, and if this is needed routinely save it as a Scheduler file and run it when you need to. E.g. if you need to export pricing and inventory information, then simply select the relevant fields: Product ID, Variant ID, Product Name, Variant Name, Inventory, Price, Sale Price (if you need to, you can also add the MSRP, COST, SKU and or SKU SUFFIX to the mix) this will give you a condensed file to work with and it will be fairly quick to export.

Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

If at all possible see if you can run exports as close as you can to where your live site is hosted, and if you have the option of remotely logging into your server that hosts your site, install commander on the server where you host your site, you'll find the speed to be the fastest there.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

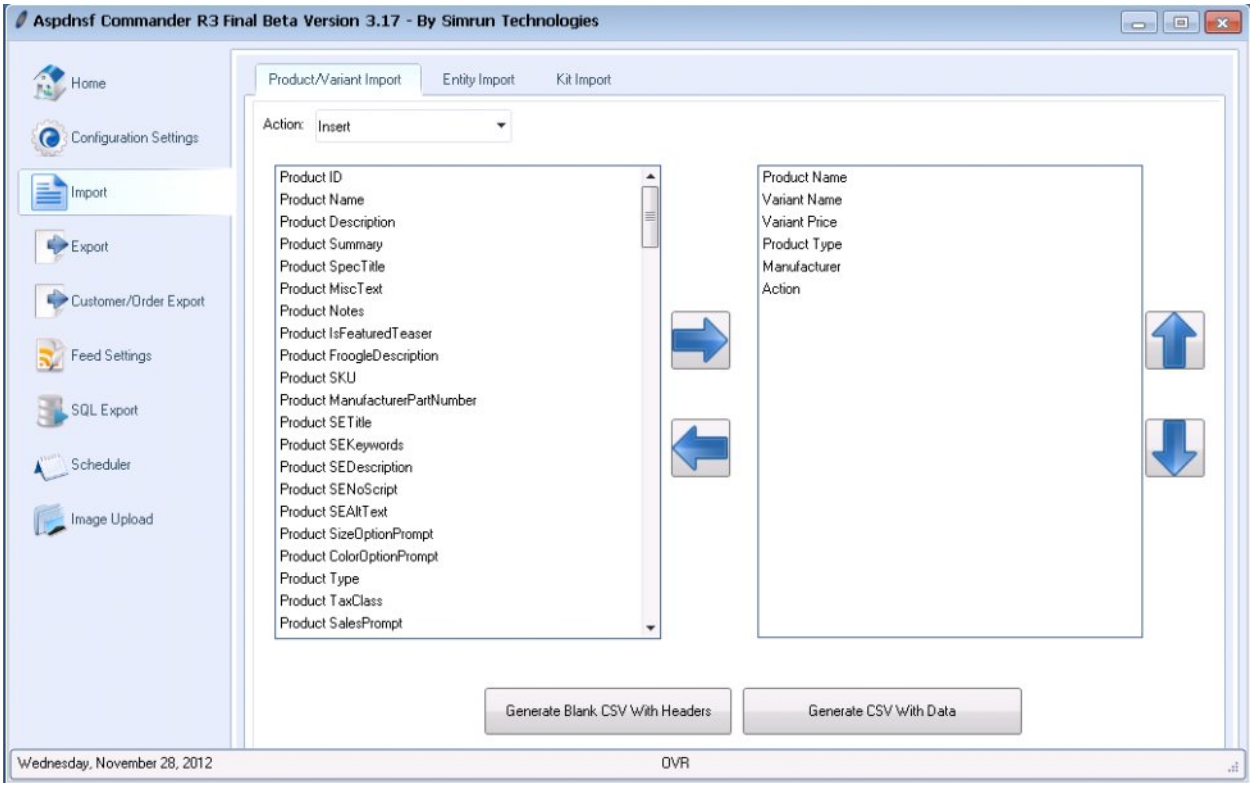
# Chapter 3: IMPORTS, UPDATES & DELTES WITH ASPDNSF COMMANDER R3

## PERFORMING PRODUCT IMPORTS

Aspdnsf Commander is the tool of choice when it comes to performing product and entity imports and updates on your Aspdotnetstorefront site. Utilizing the capabilities of the WSI platform, Aspdnsf Commander allows inserting and updating information via simple to manage .csv files.

As noted earlier, we recommend you start using Aspdnsf Commander by first performing some basic exports to familiarize yourself with the data format and field names in Aspdnsf Commander R3. This will teach you how Commander handles data and will give you a very good grasp of what you need to do to perform imports and updates.

Product imports are fairly straight forward in Aspdnsf Commander, begin by going into the import tab and select the Product/Variant Import (This should be selected by default).




Inside the Product/Variant Import tab you will see an Action drop down whit three available options:

- Insert (to add products/variants into your site)

- Update (to update product/variant information)
- Delete (to delete product/variant information)* For more information about delete please go to the Deleting products & Information section.

By selecting the action you will enable to controls on the left and right property lists, these show which options are available for you to use, each action has a minimum amount of properties that need to be added to ensure the proper result. E.g. the Insert Action requires that you provide at least the following information in your .csv file:

- Product Name
- Variant Name (this can be left blank, simply type a blank "SPACE" in your variant name)
- Variant Price
- Product Type
- Manufacture
- Action (This is the most important property and is unique to insert, updates and deletes)

While the above will get a product and its default variant inserted into your database, we recommend you add additional fields of information such as the category, display package, description, summary, SKU, and any other pertinent information into your import list before populating your csv file.

Once you have selected the fields you want to use for your product insert, you have 2 options:

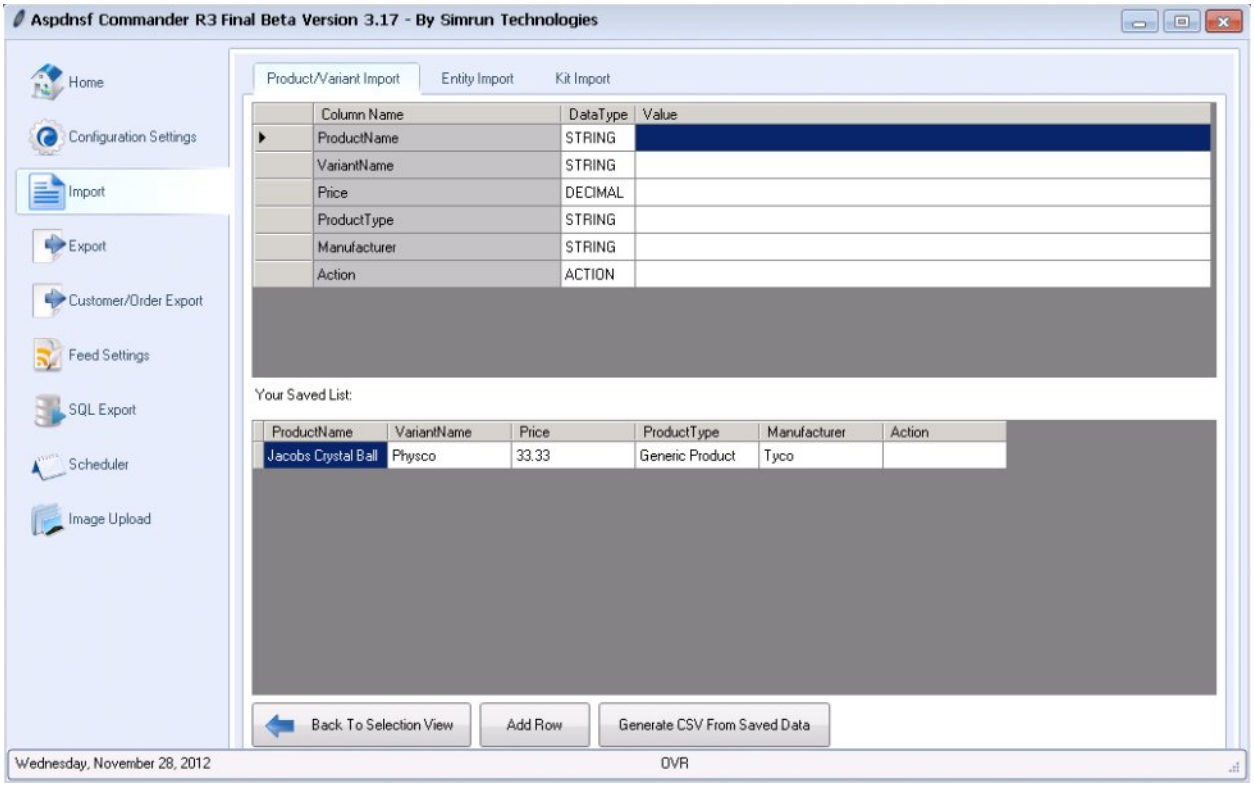
- Generate Blank CSV With Headers
- Generate CSV With Data

Essentially, at this point you are telling Aspdnsf Commander that you would like to begin populating your csv file with information that you will use to upload or insert products into your Aspdotnetstorefront site.

The Blank CSV with Headers will create a csv template for you that you can than begin inserting data into, I.E. it will create the headers for you and then you'll need to insert the information. The data format is exactly the same as the export files, with the exception of the Action Column. In the action column you'll need to specify what action you would like Aspdnsf Commander to perform; E.g. if you are performing an Insert, you'll need to type insert for each row of your CSV file. NOTE: While it is possible to multiple actions per csv file, we generally recommend you perform only 1 type of action per csv file. Click the Generate Blank CSV with Headers to have Aspdnsf Commander provide you with a csv template file with the headers.

Generate CSV with Data provides the user with a basic gui that will allow them to build their product and variant csv file by typing information into a basic grid.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.




In either case, both options will generate the same type of CSV file for you and the data format follows the same format as the exports, i.e. In the case of products and variants, each represents 1 variant (if you need to see how this looks like, please perform a sample export first before trying to import data into your site).

The GUI options will be improving with newer version R3 as time goes by, please be sure to check on the latest version of the manual for new enhancements and features.

When working with csv import / inset file it's important to consider what information and what process you would like Aspdnsf Commander to perform. It is highly recommended you add more than the minimum required files to perform an insert as including Category, Display, Sales tax and other information will be needed so that you are can see the product on your site. Imports and updates are unique in that many fields in Aspdotnetstorefront have an automatic default, such as if you don't specify the publish column your product will automatically set to isPublished = true in the database, that is the product will be available to display on your site. The same is true for various other options such as Start and End Date, Default Tax class, show Buy Button and more (see Data and Data Types for more information).

Another important option to consider is that you should use the correct Action when working with Aspdnsf Commander; that is if you are adding new products to your site, make sure you set the action to Insert and NOT Update or Delete. While Aspdnsf Commander R3 has built in protection to stop duplicates and can detect if a product already exists (ie. If you set an existing SKU that's already in your

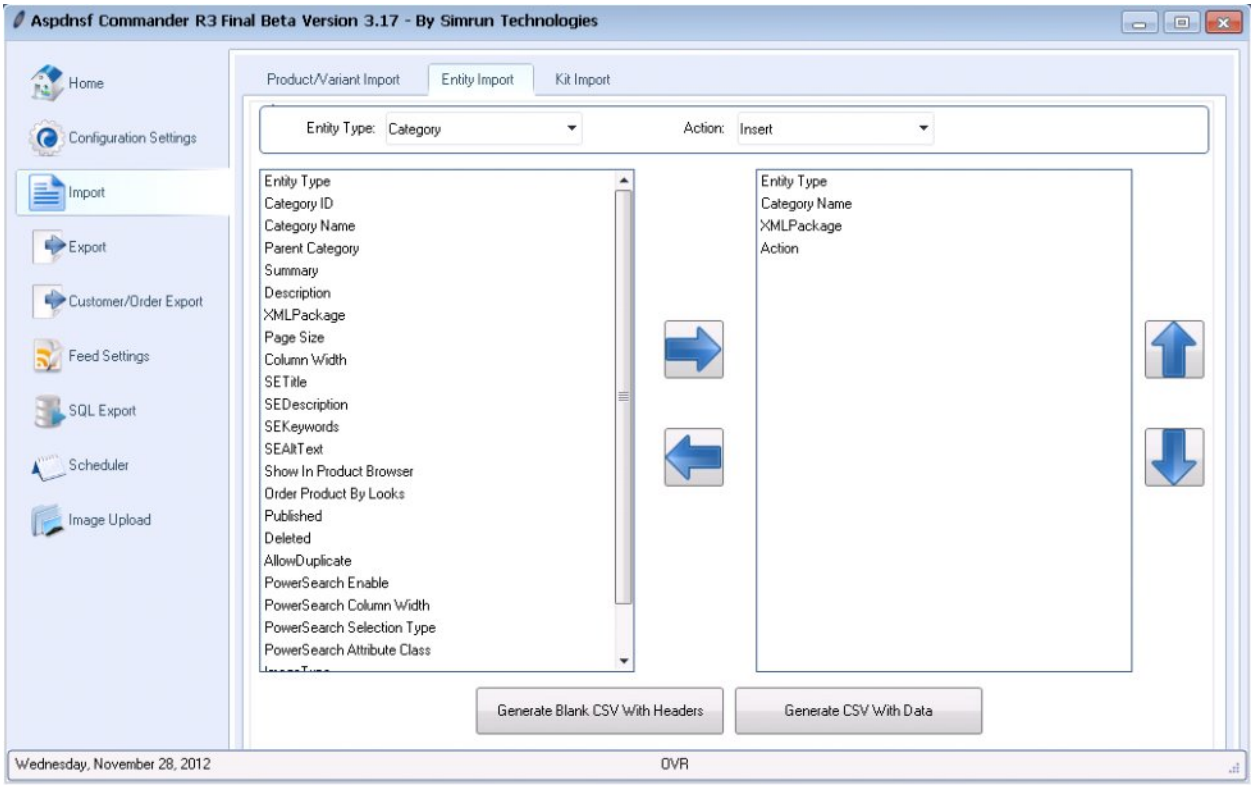
Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

database to insert in your csv file, commander R3 will perform an UPDATE instead of creating a duplicate variant/product) there are certain situations where a duplicate can occur if you use the wrong action, for more information see the Data and Data Types section of this manual.

Once you have your csv file ready it will now be time to process the file, to see how to process the file please see the SCHEDULER & PROCESSING FILES.

## IMPORTING ENTITY INFORMATION

Aspdnsf Commander R3 has dedicated import tabs for entities and includes support for categories, manufacturers and sections (departments). To start setting your csv file for entities, go to the Entity import tab and you'll see a screen similar to the product/variant Import tab. The interface is very similar to the product/variant import, however, you will first need to select the type of entity that you would like to work with and after this you will need to select the action.




Each Entity has a different number of minimum criteria, when you select the entity, you then need to select the

As with the product/variant import you'll have the option to generate the csv file with blank headers or you can generate the csv file using our basic gui interface.

NOTE: While it is possible to create entities on the fly with the product import (e.g. If you entering a product and mapping it to categories that don't exist on your site, commander will create those

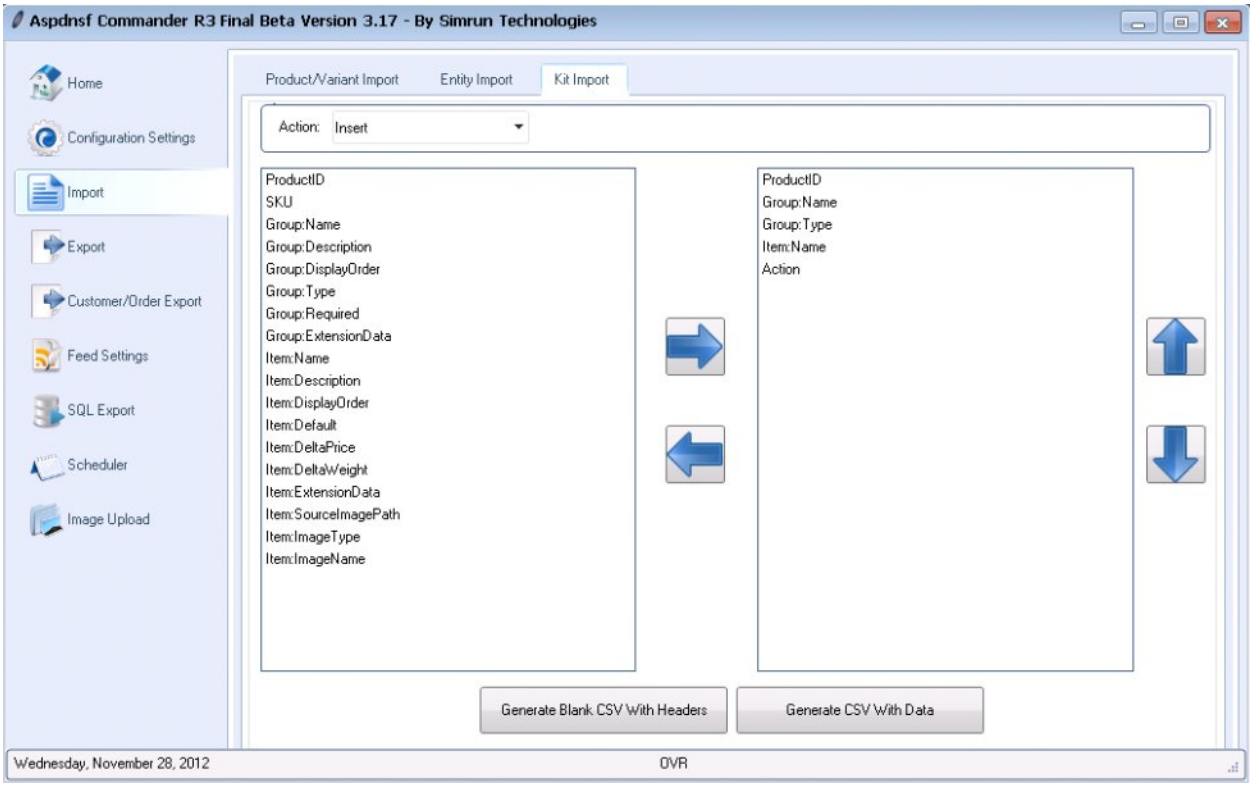
categories for you while you are importing the data) product/variant import doesn't have the option to specify all the possible values for entities, so in this case using the entity import tabs will be needed.

Also note, not all options are available for all customers, e.g. on the Category Import you may see options such as PowerSearchEnable this is an S.E.E. (Simrun Enterprise E-commerce) ONLY feature and will not work on stock Aspdotnetstorefront sites.

## IMPORTING KIT INFORMATION

Kit information is one of the unique features in Aspdnsf Commander in that it allows you to populate your product kit information into product you have already uploaded to your site. NOTE Kit import should only be used on products that you have uploaded and set to is kit = true, do not try to upload kit information on products that have not been uploaded or are not set to kits. It is also highly recommended that you first perform a kit export before attempting to import or update kit information via Aspdnsf Commander R3 as the file format is drastically different from the product/variant and entity import files.

To access the kit import options click on the Kit Import tab:




Once here, select kit information you would like to insert of update, just as in the case of products, you can chose for Aspdnsf Commander to create a blank csv file for you or you can populate the data inside a simple GUID based form.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

## PERFORMING UPDATES

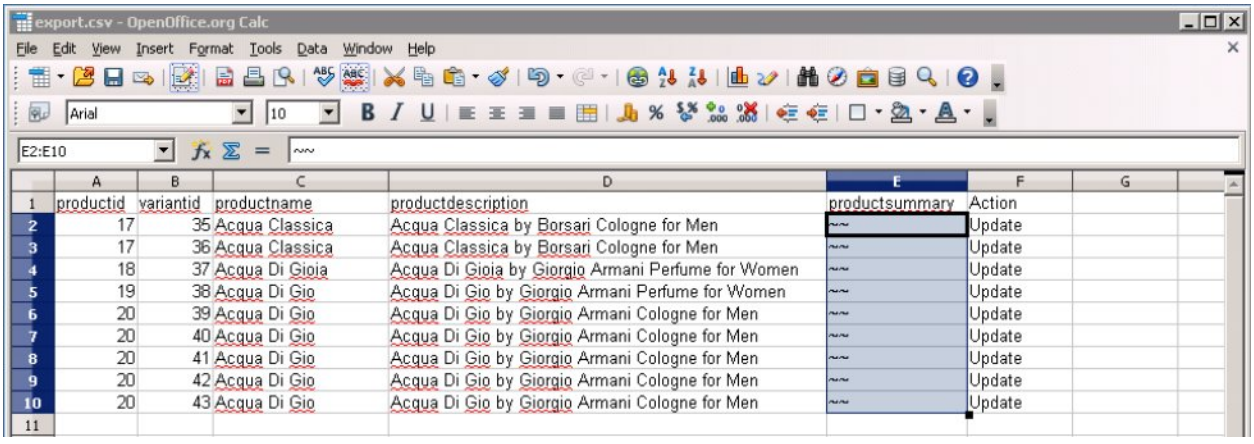
Updating information in ASPDNSF Commander R3 is very similar to performing imports, with the exception that there are generally different required fields to perform the action. In its simplest form, the major difference between inserting and updating information is that you can not specify or change IDs during updates but you can specify which ID you would like to work with. This means for example while performing and insert you can not specify a productID or variantID this is because these values don't exist in your website database, Aspdotnetstorefront will create these for you once the new data is inserted, however, during an update you can specify the product/variant ID (or if you are working with entities, the entityID) because the ids do exist in the system.

As a safety measure, Aspdnsf Commander will try its best not to create a duplicate item if you accidentally specify the Insert action instead of the update, if commander detects a duplicate SKU, it will perform an update instead of an insert for you. However, its best practice you specify the correct action.

Another important consideration is to only update the fields you need to update, e.g. If you only need to update the price and inventory, there is no need to include the fields from various other fields that don't need to be changed, including more fields will only add additional processing time and will slow the completion of the batch file and take up valuable resources.

## UPDATE PARSING

Commander will update information based on what is provided in your csv file, if you wish to "NULL" or remove a value in any column while doing an update, use the following value in your csv file to set them to NULL: ~~ the "double tilde" character is a way to tell Aspdnsf Commander R3 that you would like to remove the value for this field on this record. In Previous versions of commander we recommended typing a blank space in to remove the information, however, with R3 we have added support for NULL values as a blank space is not the same as a NULL value.




| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | productid | variantid | productname | productdescription | productsummary | Action | |
| 2 | 17 | 35 | Acqua Classica | Acqua Classica by Borsari Cologne for Men | ~~ | Update | |
| 3 | 17 | 36 | Acqua Classica | Acqua Classica by Borsari Cologne for Men | ~~ | Update | |
| 4 | 18 | 37 | Acqua Di Gioia | Acqua Di Gioia by Giorgio Armani Perfume for Women | ~~ | Update | |
| 5 | 19 | 38 | Acqua Di Gio | Acqua Di Gio by Giorgio Armani Perfume for Women | ~~ | Update | |
| 6 | 20 | 39 | Acqua Di Gio | Acqua Di Gio by Giorgio Armani Cologne for Men | ~~ | Update | |
| 7 | 20 | 40 | Acqua Di Gio | Acqua Di Gio by Giorgio Armani Cologne for Men | ~~ | Update | |
| 8 | 20 | 41 | Acqua Di Gio | Acqua Di Gio by Giorgio Armani Cologne for Men | ~~ | Update | |
| 9 | 20 | 42 | Acqua Di Gio | Acqua Di Gio by Giorgio Armani Cologne for Men | ~~ | Update | |
| 10 | 20 | 43 | Acqua Di Gio | Acqua Di Gio by Giorgio Armani Cologne for Men | ~~ | Update | |
| 11 | | | | | | | |

In the example above, the ~~ will remove values in the product summary.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

## PERFOMING DELETES

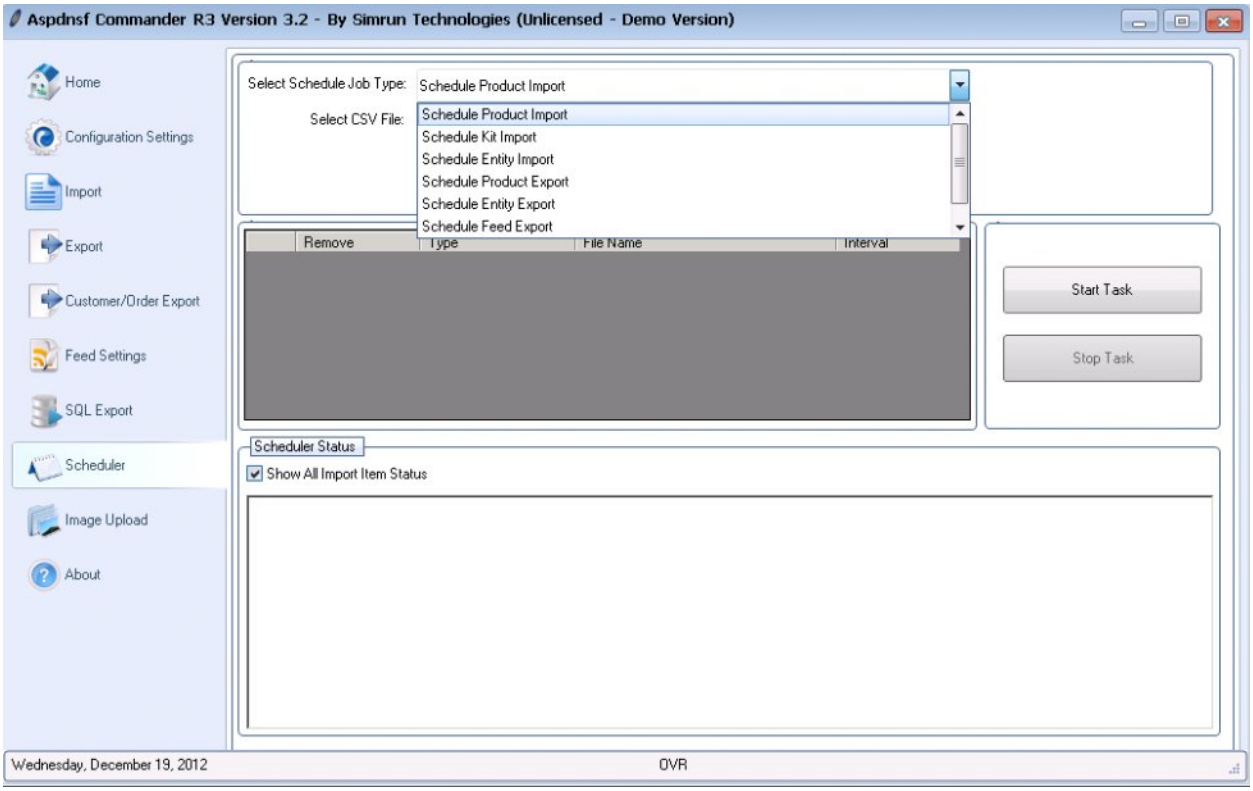
Aspdnsf Commander R3 has a fairly simple way of performing deletes; this process involves specifying the ID or SKU and specifying the action; Aspdnsf Commander R3 then remove that record (or row) from your database (in some cases such as products and entities will be set to isDeleted = true which means they are not removed from the database but are not active items in your database).

For products/variants, simply specify the productID, or SKU and then in the action set the value to delete, this will then delete the product. To delete a single variant, specify the productID, Variant ID and then the action, this will delete that single variant. Entities work in the same way; however, you will need to specify the entityID as skus are not available for entities.

For kit items, you will need to reference the kit tem or group by the name, this is because Aspdnsf Commander R3 does not support kit and kit group IDs due to the way Aspdotnetstorefront handles the kit system, If you have changed the name than you need to reference the correct name for commander to delete individual kit items or groups.

## PROCESSING YOUR IMPORT/UPDATE/DELETE FILES

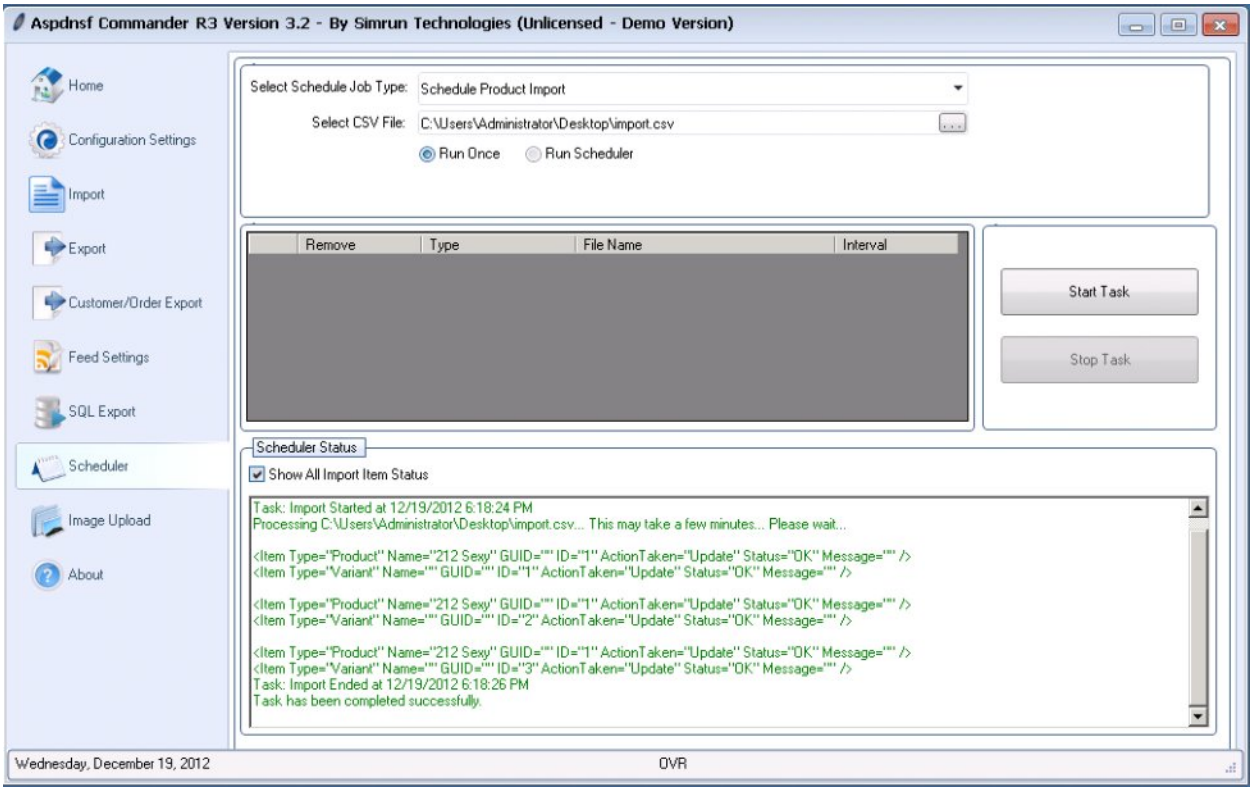
Once you have created your import, update or delete files it is now time to process these files. Unlike exports, in Commander R3 you must schedule your processes for imports/updates/deletes; this is done in the scheduler tab (For more details about the scheduler please see Chapter 4).




Copyright © 2003 - 2013 Simrun Technologies L.L.C. 

To process your file, click on the scheduler tab and from this tab you will need to select which type of file you would like to process. You'll notice you have options to process Product, Kit, and Entity Imports as well as options on performing exports (see Chapter 4 on how to schedule routine exports and imports). Select the type of import you would like to do and then click on the Select CSV FILE to select your csv file. Once you have your CSV file you will notice the RUN ONCE radio button is pre-selected for you, you will want to keep this selected if you plan on running your file immediately, if you need to run your file later, you can select the Run Scheduler (see Chapter 4 for more details). After you select your file, you can then click the Start Task button.

Upon clicking the Start Task button you will see commander is now starting to parse the csv file and will show you the import log in the Scheduler Status window, this will display the response back form the WSI component inside AspDotNetStorefront and will let you know if the records were processed successfully or not. GREEN notifications records were processed correctly, whereas RED notifications means that there was an error processing a record in your CSV file.




After the import process is complete you can view a log of the transaction inside your log folder. NOTE: Aspdnsf Commander R3 will MOVE your file form the source directory into the processed folder (set in the configuration tab) once it has processed the file, if you need to review your file after it has been processed, you can do so by going into the processed folder, the file will have a time stamp added to the name to let you know when the file was processed.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

In case of errors, or if you would like to have a log of transactions that happened, you can open up your log file and it will display all the responses it received from the WSI system during your import/update/delete operations.

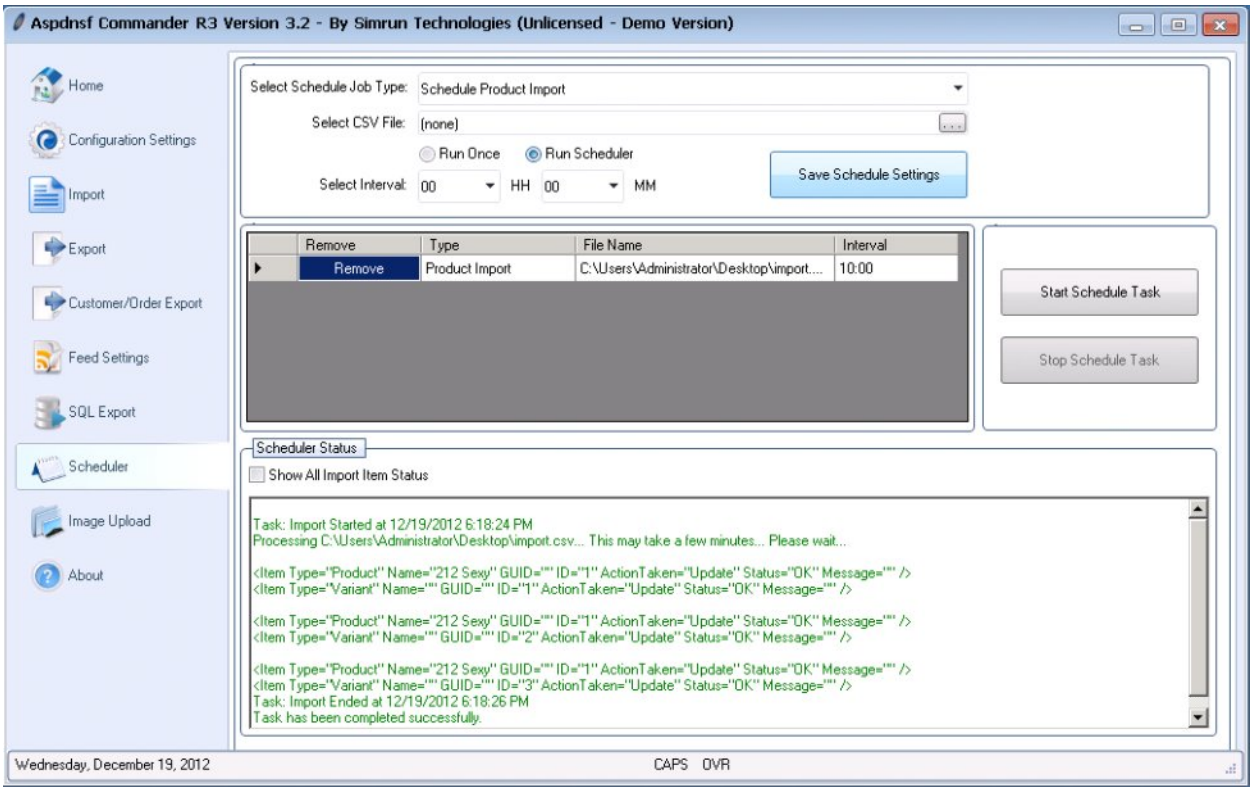
NOTE: if at any time you wish to abort your process, you can do so by clicking on the Stop Schedule Task button and this will abort the process (any changes already done to the database WILL NOT BE UNDONE however, any records left in the file will not be processed).

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

# Chapter 4: SCHEDULER

## USING SCHEDUELR TO SETUP ROUTINE TASKS

The scheduler inside Aspdnsf Commander R3 is a great way to automate routine tasks that you may have during day to day activities of managing your AspDotNetStorefront site. In essences it allows you to automatically process import/update/delete files as well as automate exports at the time interval that you wish. This comes in handy in situations where you may have a back office system or point of sale system where you need to have routine inventory/pricing updates done to you site (e.g. every 15 minutes your POS spits outs an updated inventory and pricing file which is then picked up by Aspdnsf Commander R3 and then your site is updated.) You can also schedule multiple jobs in the scheduler so you can have both import/update/delete processes as well as export processes running.




To setup a scheduled job, click the scheduler tab and then select the type of job you wish to schedule in the Select Schedule Job Type dropdown:

- Schedule Product Import: This is used to process import/update/delete files containing product and variant information, to see how to create this file please see Chapter 3.
- Schedule Kit Import: Similar to product import, please see Chapter 3 for more details

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

- Schedule Entity Import: Similar to product import but used for entities, see Chapter 3 for more details.
- Schedule Product Export: If you have created a scheduled export file, you can use this option to re-export the same fields that you have saved in your export xml configuration file, for more information please see Chapter 2
- Schedule Kit Export: Similar to product export, however, for item kits and groups.
- Schedule Product Export: Similar to product export, but used for exporting entity information.
- Schedule SQL Export: Used to export SQL queries that has been saved as an XML configuration file.

After you have selected your export type, you will now need to select the Run Scheduler option to set an interval time, this will then tell commander that you want to the scheduler to process this tasks at the interval you set (e.g. setting a 5 hour interval will mean you will want Aspdnsf Commander R3 to re-run this task every 5 hours). After selecting your interval click on the Save Schedule Settings button to add this job to the scheduler's task list.

To begin your scheduled tasks, select Start Schedule Task this will then start running the items on the task list, higher priority tasks will run first followed by other tasks listed in the task pane. NOTE: The scheduler will not begin a proceeding task before it has finished its current task, e.g. if you have an item scheduled to run at 1:00 P.M. but your current task has not finished, the proceeding task will be delayed until the first task has successfully completed. Its recommended you keep a large enough time gap between tasks so that you don't run into the situation where a scheduled task is being delayed because the current task has not finished processing.

## STATUS WINDOW & LOG FILES

As explained in the end of Chapter 3, the status window displays the progress as Aspdnsf Commander R3 processes the files, Green lines means operations completed successfully whereas Red Lines mean there was an error processing the request.

For all imports/updates/delete files Aspdnsf Commander R3 maintains a log files as well as a copy of the original csv file inside the and log and processed folders respectively. This allows administrators and operators to view what operations where performed during the import/update/delete. There are no log files created for export operations, however, the exported files are stored inside the processed folder inside a sub folder called export.

Please note, if you perform many operations you may want to keep an eye out for the processed and log file folders, these files are not compressed and may take up a large amount of space after repeated processing and use. Its recommended you backup and remove older log files from these folders to avoid losing storage space on your drives.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

## TIPS & RECOMMENDATIONS

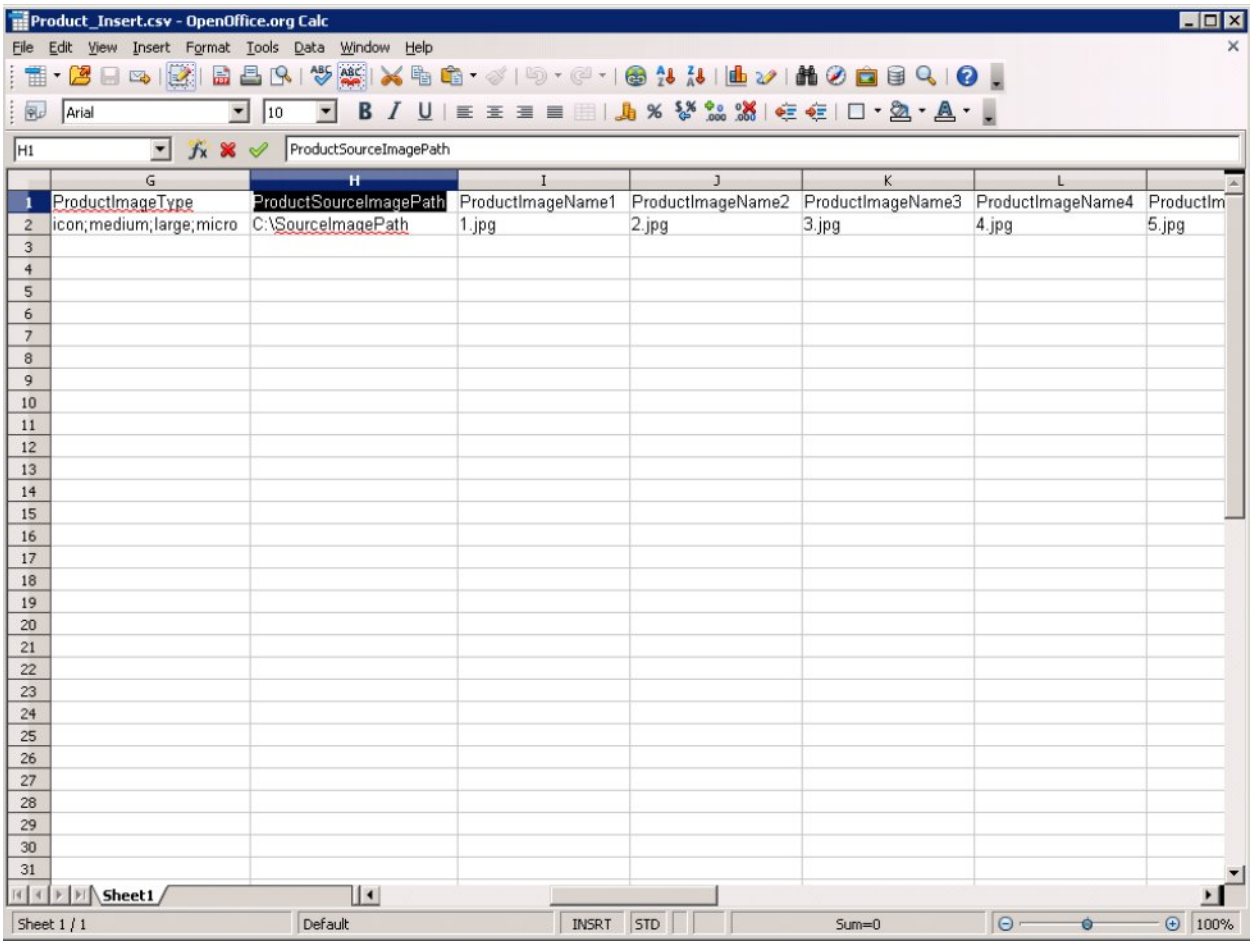
When using the scheduler, make sure you do not have files that you need to have processed upon with the your spreadsheet program as this can cause a windows permission error when Aspdnsf Commander R3 tries to move the csv file into the processed folder.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

# Chapter 5: IMAGE HANDLING

## IMAGE UPLOADING VIA CSV

In Aspdnsf Commander R3, the biggest and most significant feature that was added was the ability to upload images directly to your website without having to manually ftp them was in the case of R2. Image uploading in Commander R3 not only allows you to upload a single image/item but it allows you to upload 10 images/product as well as upload images for kits, variants and also has the ability to automatically resize and shape images based on the values that you setup in the appconfigs inside your Aspdotnetstorefront Admin.




The fastest way to upload images is to use the CSV file method, which essentially allows you to specify multiple images / product and variants and then uploads these images into your Aspdotnetstorefront website. NOTE: You must make sure you configuration is setup correctly and that the FTP credentials are working, you'll need to make sure you have both write and read access. In Aspdnsf Commander R3 we have added new columns which will allow you to upload images, these are:

- **ProductImageType** valid values are: icon,medium,large, and micro (separated by semi-colon) The Product Image Type tells Aspdnsf Commander R3 which sizes of images need to be created and FTP'd to your Aspdotnetstorefront site, its best practice to create all 4 available sizes.

- **ProductSourceImagePath** this column specifies the location of the image files on your client (your computer that has the images), you will need to make sure all the images you plan on uploading are in the directory you specify.

- ProductImageName1, ProductImageName 2 (e.t.c.) these columns represent the names of the images that need to be uploaded, here you can use as many columns as needed (up to 10). Aspdnsf Commander R3 allows you to specify up to 10 images/product and these will be automatically resized and uploaded for you. Please note, the images will be renamed to AspDotNetStorefront's normal image naming convention upon being uploaded, e.g. X_1_.jpg, x_2_.jpg where X represents the product ID assigned to the product.

For variants, use the variant columns (see chapter 6 for column names or the included sample csv files), the same is true for kit items and groups.

Once you have entered in relevant image information, process the file as you normally would with the scheduler.

Support for entities is also available; please see chapter 6 or the sample csv files to familiarize yourself with the column names and format.

Please note, ImageFileNameOverride will not resize and upload images, this is used as it has always been used in Aspdotnetstorefront and is a column that references the image file in image directories. If you are using ImageFileNameOverride than you will need to resize and manually upload these images; this may change with a future release of R3 but as of 3.20, ImageFileNameOverride will not be resized or uploaded by Aspdnsf Commander R3.

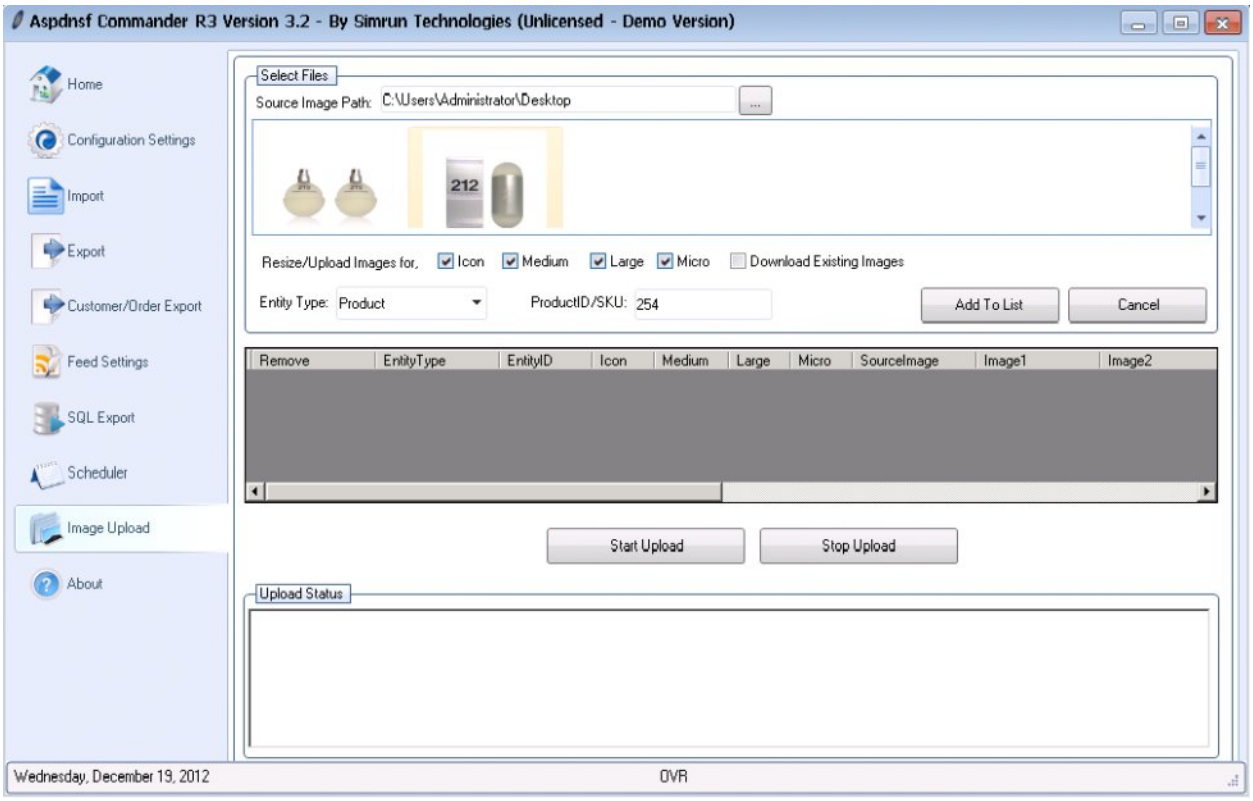
## IMAGE UPLOADING IN THE IMAGES TAB

Aspdnsf Commander R3 also allows you to upload the images via an image upload tab which provides a GUI and drag and drop functionality to allow you to mass upload images for products, entities as well as kits.

If you plan on using the gui, simply click on the Image Upload Tab and then select your images by the Source Image Path: browse box; you can alternatively drag and drop images into the Aspdnsf Commander R3 image area. You can change the order of the images by dragging the images back and forth to the order that you would like, this will allow you to set the default image as needed.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com




Once you have your images added you can then tell Aspdnsf Commander R3 what image sizes you would like to have created, by selecting the Icon, Medium, Large, and Micro sizes (its best practice to select all of the sizes as this will insure all relevant images are uploaded, please note all types of images are supported for each type of upload e.g. MICRO images are not supported by entities and kits). The next step is to enter the ID (or in case of products either ID or SKU) for the image that is being uploaded, the ID will be used to rename the file during re-sizing and upload. Note: If you need to see which images are currently being used for the product, put a check on the Download Existing Images checkbox and this will download and display the images currently being used by that particular entity, product, or kit item.

After you have set your images, simply click on the Add to List button and this will add the image to the task list, you can add multiple items to the task list and when you are ready to process all of you images, click the Start Upload button. The Upload Status window will show you the results, NOTE: you can also view these same results inside the log files in your log folder (see configuration/Chapter 1 for more details)

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

# Chapter 6: DATA & DATA TYPES

## PRODUCT/VARIANT CSV DATA AND COLUMN NAMES

The table below lists the allowed data and the type of data for each of the columns (properties) for handing product and product variant information. The notes section explains any precautions or limitations associated with the particular property (column).

| Column Name | Data Type | Example | Description | Notes |
|---|---|---|---|---|
| ProductName | Text | Test Product | It defines the name of the product. | |
| VariantName | Text | Test Variant | It defines the name of the product's variant. | |
| Price | Numeric | 10.00 | It defines the price of the product's variant. | Do not add "$" or any other currency signs. |
| ProductType | Text | Generic Product | It defines the product type (e.g. generic, gift card, kit, etc). | |
| Manufacturer | Text | MfgName or \MfgName | It defines the predefine manufacturer name. | Use backslash to setup sub manufacturers |
| Action | Text | Insert/Update/Delete | If Insert: Product should be inserted into the master table.<br><br>If Update: Product should be updated according to value in Entity Type and ProductID columns.<br><br>If Delete: Product should be deleted according to value in Entity Type and ProductID columns. | |
| ProductID | Numeric | 12(existing productid in database) | It should be existing ProductID in database for Modifying or Deleting Product when Action Column value is "Update/Delete".<br><br>Note: When Action Column value is "Insert" i.e. when inserting the new product into the database this column is not required, i.e. remove this column. | |
| ProductDescription | Text / HTML | Product Description | It defines the product description. It should be either simple text or HTML content. | Can be simple text or HTML (must be valid XHTML) |
| ProductSummary | Text | Product Summary Text | It should be either simple text or HTML content. It is the short details of the product. | |
| SpecTitle | Text | Click here | Spec title is the actual link text that will show up on | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| | | | your product page. This should say something like “Click here!” | |
| MiscText | Text | Simple Text | Misc Text as shown on the product page. | |
| ProductNotes | Text / HTML | This is a Text Note | It defines the notes related to product. | |
| IsFeaturedTeaser | Boolean (0/1) | 0 or 1 | Optional, if left blank it defaults to 0 | |
| ProductFroogleDescription | Text / HTML | Froogle Desc Text | Used for Product Feeds | No HTML is allowed. |
| SKU | Text | ABC-123 | It defines the product SKU. | No Special Characters e.g. “#” |
| ProductManufacturerPartNumber | Text | 123F456 | It defines the product manufacturer part number. This value will be sent in distributor emails if populated (optional) | No Special Characters e.g. “#” |
| ProductSETitle | Text | Simple Text | It is used for SEO purpose. | No Html |
| ProductSEKeywords | Text | Simple Text Separated By Comma | It is used for SEO purpose. | No Html |
| ProductSEDescription | Text / HTML | Html Text Or Simple Text | It is used for SEO purpose. | No Html |
| ProductSENoScript | Text | Simple Text | Message displayed if client browser has JavaScript turned off. | No Html |
| ProductSEAltText | Text | Simple Text to be displayed on image Hover | It is used for SEO purpose. | No Html |
| SizeOptionPrompt | Text | Product Size: | It defines text to display to customers for attributes they may choose instead of "Size:" | No Html |
| ColorOptionPrompt | Text | Product Color: | It defines text to display to customers for attributes they may choose instead of "Color:” | No Html |
| TaxClass | Text | Goods, Alcohol, Shipping etc.. | It defines which tax class to use to look up/apply tax rates to this product during checkout. | Enter in the TAX Class NAME not the ID |
| SalesPrompt | Text | On Sale, Sale For etc. | If your product has a sale price set, this determines which prompt precedes that sale price. | Type in the prompt NAME not the prompt ID |
| SpecCall | Text | /moreinfo/productspec/37.html | Path of the Spec File. | |
| SpecsInline | Numeric | 0 or 1 | If the ‘Specs Inline’ value is set to 1, the html file or topic will appear directly on the product page otherwise it should be popup in another window. | |
| XmlPackage | Text | product.seeproductpage.ascx | It should be valid XMLPackage Name. According to XMLPackage Product page will be rendering. | |
| ProductImageFilenameOverride | Text | Image Name, if you want to Refer your image by some Specific Name else leave blank | It allows you to specify a filename for the software to use for this product's images (by default, images are saved as productID.jpg). | NOTE: this field is NOT uploaded; if you want the images to be uploaded please see the Image Upload column names. |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| ColWidth | Numeric | e.g. 4 | For display formats that use a grid. It defines how many columns are created before creating a new row. In this example 4 columns will be create and next product displayed in next row. | |
| PageSize | Numeric | e.g. 20 | Determines how many sub-Category or Products are displayed on the page while browsing the entity. In this case 20 Sub-Categories/Products should display while browsing, If It is more than 20 then paging should be appeared. | |
| SkinID | Numeric | Skin_1 | It defines the valid skin name. | |
| TemplateName | Text | Template1 | Enter the Template name if you want to Load the Product Page in some Different Template Page. | |
| RelatedProducts | Text | 36,37,42,56 | Enter comma separated ProductIDs (Not Name, not Sku). | |
| UpsellProducts | Text | 37,41 | Enter comma separated ProductIDs(Not Name). | |
| RequiresProducts | Text | 37,38,40 | Enter comma separated Product IDs (Not Name). If a customer adds this product to the cart, the required product(s) will be added as well. | |
| IsFeatured | Boolean | 0 Or 1 | 0 = False, 1 = True | If left blank the system will default to false |
| IsSystem | Boolean | 0 Or 1 | 0 = False, 1 = True | If left blank the system will default to false |
| IsAPack | Boolean | 0 Or 1 | If 1 then it should be product as a Pack, It allows you to group items that you normally sell separately into one item. | If left blank the system will default to false |
| PackSize | Numeric | Numeric Value e.g. 10 | It defines the number of products you wish the pack to contain. | |
| ShowInProductBrowser | Boolean (0/1) | 0 Or 1 | This determines whether or not the product will appear in the browser. 0-No, 1-Yes. | If left blank the system will default to false |
| ShowBuyButton | Boolean (0/1) | 0 Or 1 | If 0 then customer will not be able to add this item to cart. | If left blank the system will default to true |
| ProductPublished | Boolean (0/1) | 0 Or 1 | 0-No, 1-Yes. This option determines whether or not the category will appear in the menus/sitemap. | If left blank the system will default to true |
| ProductWholesale | Boolean (0/1) | 0 Or 1 | 0 i.e. false or 1 i.e. true | If left blank the system will default to false |
| RequiresRegistration | Boolean (0/1) | 0 Or 1 | If 1 then it hides the product from anonymous customers. | If left blank the system will default to false |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| HidePriceUntilCart | Boolean (0/1) | 0 Or 1 | If 1 then it prevents the item price from showing up on the product detail page. | If left blank the system will default to false |
| IsCalltoOrder | Boolean (0/1) | 0 Or 1 | If 1 then a "Call to Order" prompt is displayed to customers, and there are no "buy" buttons. | If left blank the system will default to false |
| ExcludeFromPriceFeeds | Boolean (0/1) | 0 Or 1 | If 1 then it prevents the software from exporting this product's info when feeds (PriceGrabber, Google, etc) are executed. | If left blank the system will default to false |
| GoogleCheckoutAllowed | Boolean (0/1) | 0 Or 1 | It determines whether customers can use the Google checkout button on the cart page (if configured) or must go through standard checkout. If 1 then Google checkout button should be appeared. | If left blank the system will default to true |
| RequiresTextOption | Boolean (0/1) | 0 Or 1 | If 1 then customers to enter personalization/custom information on product detail page. | If left blank the system will default to false |
| TextOptionMaxLength | Numeric | Numeric Value e.g. 100 | Max number of characters allowed in the text field | If left blank the system will not restrict lenght |
| TextOptionPrompt | Text | Enter Your Detail | If 1 then It defines that how to label the text box. | |
| AvailableStartDate | Date | Date e.g. yyyy-MM-dd | The first date you wish the product to appear to customers. | |
| AvailableStopDate | Date | Date e.g. yyyy-MM-dd | The last day you wish customers to be able to view that product. If you don't define stop date then left with blank. | |
| ProductExtensionData | Text | HTML Or Simple Text | It defines the internal notes, special instructions, special extra data etc. | |
| ProductExtensionData2 | Text | HTML Or Simple Text | // | |
| ProductExtensionData3 | Text | HTML Or Simple Text | // | |
| ProductExtensionData4 | Text | HTML Or Simple Text | // | |
| ProductExtensionData5 | Text | HTML Or Simple Text | // | |
| ProductExtensionData6 | Text | HTML Or Simple Text | // | |
| ProductExtensionData7 | Text | HTML Or Simple Text | // | |
| ProductExtensionData8 | Text | HTML Or Simple Text | // | |
| ProductExtensionData9 | Text | HTML Or Simple Text | // | |
| ProductExtensionData10 | Text | HTML Or Simple Text | // | |
| ProductImageType | Text | e.g. icon;medium;large;micro | Type of Image should be define with semicolon (;) separation. | Select the sizes of images you would like to have created. |
| ProductSourceImagePath | Text | C:\Path | It defines the valid path of the source images. | |
| ProductImageName1 | Text | 1.jpg | It defines the valid name of the image. File extension should be either ".jpeg/.jpg" or ".gif" or ".png" | Uploads the image as Imag1 as well as the default image. |
| ProductImageName2 | Text | 2.jpg | // | // |
| ProductImageName3 | Text | 3.jpg | // | // |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

7902 Wrenwood Blvd Suite B · Baton Rouge, LA 70809 · 1-877-SIMRUN1 · www.simrun.com

| | | | | |
|---|---|---|---|---|
| ProductImageName4 | Text | 4.jpg | // | // |
| ProductImageName5 | Text | 5.jpg | // | // |
| ProductImageName6 | Text | 6.jpg | // | // |
| ProductImageName7 | Text | 7.jpg | // | // |
| ProductImageName8 | Text | 8.jpg | // | // |
| ProductImageName9 | Text | 9.jpg | // | // |
| ProductImageName10 | Text | 10.jpg | | // |
| Category1 | Text | \ParentCate\Child | User can map the product with category hierarchy. If category hierarchy is not present in master table then first it creates the category hierarchy (insert categories) and map product with that category. | In order to map products to multiple categories, separate the paths with a semi-colon ";" |
| Category2 | Text | \ParentCate\Child | // | // |
| Category3 | Text | \ParentCate\Child | // | // |
| Category4 | Text | \ParentCate\Child | // | // |
| Category5 | Text | \ParentCate\Child | // | // |
| Category6 | Text | \ParentCate\Child | // | // |
| Category7 | Text | \ParentCate\Child | // | // |
| Category8 | Text | \ParentCate\Child | // | // |
| Category9 | Text | \ParentCate\Child | // | // |
| Category10 | Text | \ParentCate\Child | // | // |
| Section1 | Text | \ParentSection\Child | // | // |
| Section2 | Text | \ParentSection\Child | // | // |
| Section3 | Text | \ParentSection\Child | // | // |
| Section4 | Text | \ParentSection\Child | // | // |
| Section5 | Text | \ParentSection\Child | // | // |
| Section6 | Text | \ParentSection\Child | // | // |
| Section7 | Text | \ParentSection\Child | // | // |
| Section8 | Text | \ParentSection\Child | // | // |
| Section9 | Text | \ParentSection\Child | // | // |
| Section10 | Text | \ParentSection\Child | // | // |
| Vector | Text | \Vector1;\Vector2 | User can map the product with vector hierarchy. If vector hierarchy is not present in master table then first it creates the vector hierarchy (insert vectors) and map product with that vector. | NOTE: This is not an officially supported field |
| Distributor | Text | \Distributor1;\Distributor2 | User can map the product with distributor hierarchy. If distributor hierarchy is not present in master table then first it creates the distributor hierarchy (insert distributor) and map product with that distributor. | NOTE: This is not an officially supported field |
| Genre | Text | \Genre1;\Genre2 | User can map the product with Genre hierarchy. If Genre hierarchy is not | NOTE: This is not an officially supported field |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| | | | present in master table then first it creates the Genre hierarchy (insert Genres) and map product with that Genre. | |
| ProductClass | Text | Class Name valid only for SEE DB | Class Name valid only for SEE DB | ONLY USED FOR SEE CODE BASE |
| VariantID | Numeric | 1 | It sholuld be existing VeriantID in database for Modifying or Deleting Variant when Action Column value is "Update/Delete".<br><br>Note: When Action Column value is "Insert" i.e. when inserting the new product in master table this column is not required. | This ID is generated by Aspdotnetstorefront. |
| IsDefault | Boolean (1/0) | 0 Or 1 | 1 means default variant for particular product. It should be only one default variant item for each Product Name | |
| VariantDescription | Text / HTML | HTML Or Simple Text | It defines the description of the variant. | No Html |
| VariantSEKeywords | Text | Comma Separated Keywords | It is used for SEO purpose. | No Html |
| VariantSEDescription | Text / HTML | Simple Text | It is used for SEO purpose. | No Html |
| VariantFroogleDescription | Text | Simple Text | It is used for SEO purpose. | |
| SKUSuffix | Text | e.g. 4569G | This value should be appended to the parent product SKU on receipts and during checkout. | |
| VariantManufacturerPartNumber | Text | 4852GH | It defines the variant manufacturer part number. | |
| SalePrice | Numeric | 10.00 | It defines the sale price of the product (product's variant) in the product detail page. | No $ sign |
| Weight | Numeric | 0.5 | It defines variant weight, used for shipping calculations. | |
| MSRP | Numeric | 25.00 | It defines the *manufacturer's suggested retail price(MSRP)* of the product variant | No $ sign |
| Cost | Numeric | 35.00 | It defines the cost of the product variant. | No $ sign |
| Points | Numeric | 1 | It defines the product variant point. | |
| Dimensions | Text | Length x Width x Height e.g. (25x25x30) | It defines the dimensions of the product variant. | |
| Inventory | Numeric | e.g. 555 | It defines the number of this item on hand for that product variant. | |
| VariantDisplayOrder | Numeric | Numeric Order | It defines the order of the product variant. | |
| VariantNotes | Text / HTML | HTML Or simple Text | It defines the important note of the product variant. | |
| IsTaxable | Boolean (0/1) | 0 Or 1 | It defines whether or not this variant is taxed under the class set for the parent | 0 = False, 1 = True |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| | | | product. If 1 then yes. | |
| IsShipSeparately | Boolean (0/1) | 0 Or 1 | If using realtime shipping, this instructs the carrier to calculate the cost of shipping this item in its own box. This is done by sending the product dimensions, instead of just the weight that is usually sent to the carrier. | 0 = False, 1 = True |
| IsDownload | Boolean (0/1) | 0 Or 1 | If it defines with 1 then it specifies that product variant should be downloadable. | 0 = False, 1 = True |
| DownloadLocation | Text | Location Path | If IsDownload is set to 1 then specify the path of the product variant files. | |
| FreeShipping | Boolean (0/1) | 0 Or 1 | If it defines with 1 then the item gets free shipping, or if it doesn't even need shipping at all (such as a service or digital product) | 0 = False, 1 = True |
| VariantPublished | Boolean (0/1) | 0 Or 1 | 0-No, 1-Yes. This option determines whether or not the product variant will publish. | 0 = False, 1 = True |
| VariantWholesale | Boolean (0/1) | 0 Or 1 | If 1 then yes (true) | 0 = False, 1 = True |
| RestrictedQuantities | Text | Comma separated Queantities | If your variant can only be purchased in certain quantities, list them here (ie: If you sell by the dozen – 12, 24, 36, etc) | |
| MinimumQuantity | Numeric | Enter the Min Qty which Customer Needs to purchase | Smallest quantity customers are allowed to add to the cart | |
| CustomerEntersPrice | Boolean (0/1) | 0 Or 1 | If this is 1 then customers can set their own price for the product (often used for donation or gift card products). | 0 = False, 1 = True |
| VariantExtensionData | Text | HTML Or Text | It defines the internal notes, special instructions, special extra data etc. for product variant. | |
| VariantExtensionData2 | Text | HTML Or Text | | |
| VariantExtensionData3 | Text | HTML Or Text | | |
| VariantExtensionData4 | Text | HTML Or Text | | |
| VariantExtensionData5 | Text | HTML Or Text | | |
| VariantExtensionData6 | Text | HTML Or Text | | NOT an official field but created by Aspdnsf Commander R3 |
| VariantExtensionData7 | Text | HTML Or Text | | // |
| VariantExtensionData8 | Text | HTML Or Text | | // |
| VariantExtensionData9 | Text | HTML Or Text | | // |
| VariantExtensionData10 | Text | HTML Or Text | | // |
| Colors | Text | Black[+10], Green[-10], Red | You can also specify price modifiers for these attributes by enclosing them in brackets | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| | | | immediately after the attribute name. | |
| Sizes | Text | Small[-5], Medium, Large | You can also specify price modifiers for these attributes by enclosing them in brackets immediately after the attribute name | |
| VariantImageFilenameOverride | Text | Image Name, if you want to Refer your image by some Specific Name else leave blank | It allows you to specify a filename for the software to use for this product's images (by default, images are saved as productID.jpg). | This does not upload any images, if you need to upload an image use image1, image2, image3 e.t.c. |
| VariantImageType | Text | Icon;medium;large | Type of Image should be define with semicolon (;) separation. | |
| VariantSourceImagePath | Text | C:\Path | It defines the valid path of the source images. | |
| VariantImageName | Text | 1.jpg | It defines the valid name of the image. File extension should be either ".jpeg/.jpg" or ".gif" or ".png" | |
| VariantCondition | Boolean (0/1) | 0 Or 1 | Used for Google Product Upload | 0 = False, 1 = True |
| VariantUPCCode | Text | 1234567890123 | It defines the product variant UPC code. | |
| VariantEANCode | Text | 1234567890123 | It defines the product variant EAN code. | |
| VariantJANCode | Text | 1234567890123 | It defines the product variant JAN code. | |
| VariantISBNCode | Text | 1234567890123 | It defines the product variant ISBN code. | |
| DiscountTableName | Text | 5% Discount | It defines the discount on a product. | |
| DiscountType | Text | Fixed Amount Or Percentage | It defines the type of the discount either fix amount or percentage, according to type discount amount should be calculated. | |
| DiscountValues | Text | 10-100:5.00;101-200:10.00 | It defines the discount on product according to range of amount.<br>Format :<br>[minQty1] – [maxQty1] : [DiscountAmount1] ;<br>[minQty2] – [maxQty2] : [DiscountAmount2]; … | |
| GoogleProductCategory | Text | Category1 -> Category2 | Enter the Category Which you want to upload on the Google Feed. | |
| ExtendedPrice:CustomerLevelName | Numeric | ExtendedPrice:WholeSale1<br>Value Should be<br>e.g. 10.00<br><br>You can Enter multiple Column for different Customer Levels. | If you want to Specify the Extended Price based on the Customer Level then you can Simple Create the Column in the CSV File e.g. ExtendedPrice:Wholesale1. You can create as much as columns as per the Different Customer Level.<br><br>The Value for this Column Contains the Extended Price | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| | | | (Numeric) for the Particular Customer Level. E.g. 11.00 | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

## ENTITY (CATEGORY & SECTION) CSV DATA AND COLUMN NAMES

The table below lists the allowed data and the type of data for each of the columns (properties) for Entities. The notes section explains any precautions or limitations associated with the particular property (column).

| Column Name | Type | Example | Description | Notes |
|---|---|---|---|---|
| EntityType | Text | "Category". | It should be fixed value string. | |
| CategoryName | Text | Category-1 | Name of the Category should be written here. | |
| XMLPackage | Text | entity.seecategorypagesingle.ascx | It should be valid XMLPackage Name.<br>According to XMLPackage category page will be rendering. | |
| Action | Text | It sholuld be either "Insert" or "Update" or "Delete" | If Insert: Category should be inserted into the master table.<br><br>If Update: Category should be updated according to value in Entity Type and CategoryID columns.<br><br>If Delete: Category should be deleted according to value in Entity Type and CategoryID columns. | |
| CategoryID | Numeric | 12(existing categoryid in database) | It should be existing CategoryID in database for Modifying or Deleting Category when Action Column value is "Update/Delete".<br><br>Note: When Action Column value is "Insert" i.e. when inserting the new category in master table this column is not required, i.e. remove this column. | |
| ParentCategory | Text | Category Xpath e.g. \Parent Cate.\ Child1 | User can define the category hierarchy.<br>For e.g. \Parent Cate-1\Child1, commander will check if the Parent Cate-1 is present in master database, if present then "Child1" is inserted under the "Parent Cate-1" else "Parent Cate-1" will insert first and then "Child1" insert under it. | |
| Summary | Text / HTML |   Summary of Category …. | It should be either simple text or HTML content. It is the short details of the Category. | |
| Description | Text / HTML |   Category Description …. | It should be either simple text or HTML content. It defines the category description. | |
| PageSize | Numeric | 20 | Determines how many sub-Category or Products are displayed on the page while browsing the entity. In this case 20 Sub-Categories/Products should | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| | | | display while browsing, If It is more than 20 then paging should be appeared. | |
| ColumnWidth | Numeric | 4 | For display formats that use a grid. It defines how many columns are created before creating a new row. In this example 4 columns will be create and next product displayed in next row. | |
| SETitle | Text | Search Engine Title | It is used for SEO purpose. | |
| SEDescription | Text | Search Engine Description | It is used for SEO purpose. | |
| SEKeywords | Text | Search Engine Keywords | It is used for SEO purpose. | |
| SEAltText | Text | Search Alter Text | It is used for SEO purpose. | |
| ShowInProductBrowser | Boolean | 0 or 1 | This determines whether or not the category will appear in the browser. 0-No, 1- Yes. | |
| OrderProductByLooks | Boolean | 0 or 1 | If 1 then most looks product displayed first. | |
| Published | Boolean | 0 or 1<br>If 0 then Category will not appeared when browsing | 0-No, 1-Yes. This option determines whether or not the category will appear in the menus/sitemap. | |
| Deleted | Boolean | 0 or 1 | 0 means record is not deleted, 1 means record is deleted. | |
| AllowDuplicate | Boolean | 0 or 1 | 0 means not allow creating a duplicate entry with category name. | |
| PowerSearchEnable | Boolean | 0 or 1 | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer and if the Value = 1, the Attribute Filter will be displayed in Category Page otherwise not. | |
| PowerSearchColumnWidth | Numeric | e.g. : 4 | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer then you can specify the Number of Column per row to be displayed in the Category Grid View. | |
| PowerSearchSelectionType | Text | Power Search Selection Type. e.g. : Single | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer then you need to Specify the SelectionType = "Single" | |
| PowerSearchAttributeClass | Text | Power Search Attribute Class. e.g. : Attribute Class Name | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer then you need to Specify the ProductClass Name which is belongs to the Attributes. | |
| ImageType | Text | Type of Image. e.g. : Icon;Medium;Large | Type of Image should be define with semicolon (;) separation. | |
| SourceImagePath | Text | Image Path, e.g. : C:\Image Path | It defines the valid path of the source images. | |
| ImageName | Text | Image Name e.g. : Image1.jpeg | It defines the valid name of the image. File extension should be either “.jpeg/.jpg” or ”.gif” or “.png” | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

Note: When Action is “Delete” the only required fields are CategoryID and the Action Column.

## ENTITY (MANUFACTURER) CSV DATA AND COLUMN NAMES

The table below lists the allowed data and the type of data for each of the columns (properties) for Entities. The notes section explains any precautions or limitations associated with the particular property (column).

| Column Name | Type | Example | Description | Notes |
|---|---|---|---|---|
| EntityType | Text | "Manufacturer" | It should be fixed value string. | |
| ManufacturerName | Text | Manufacturer-1 | Name of the Manufacturer should be written here. | |
| XMLPackage | Text | entity.seecategorypage single.ascx | It should be valid XMLPackage Name.<br>According to XMLPackage category page will be rendering. | |
| Action | Text | It sholuld be either "Insert" or "Update" or "Delete" | If Insert: Manufacturer should be inserted into the master table.<br><br>If Update: Manufacturer should be updated according to value in Entity Type and ManufacturerID columns.<br><br>If Delete: Category should be deleted according to value in Entity Type and ManufacturerID columns. | |
| ManufacturerID | Numeric | 12(existing manufacturerid in database) | It sholuld be existing ManufacturerID in database for Modifying or Deleting Manufacturer when Action Column value is “Update/Delete”.<br><br>Note: When Action Column value is “Insert” i.e. when inserting the new manufacturer in master table this column is not required, i.e. remove this column. | |
| ParentManufacturer | Text | Manufacturer Xpath e.g. \Parent Mfg.\ Child1 | User can define the Manufacturer hierarchy.<br>For e.g. \Parent Mfg.\Child1, commander will check if the “Parent Mfg.” is present in master database, if present then “Child1” is inserted under the “Parent Mfg.” else “Parent Mfg.” will insert first and then “Child1” insert under it. | |
| Summary | Text / HTML |   Summary of Manufacturer …. | It should be either simple text or HTML content. It is the short details of the Manufacturer. | |
| Description | Text / HTML |   Manufacturer Description …. | It should be either simple text or HTML content. It defines the manufacturer description. | |
| PageSize | Numeric | 20 | Determines how many products are displayed on the page while browsing the entity. In this case 20 products should display while browsing, If It is more than 20 then paging should be appeared. | |
| ColumnWidth | Numeric | 4 | For display formats that use a grid. It defines how many columns are created before creating a new row. In this example 4 columns will be create and next product displayed in next row. | |
| SETitle | Text | Search Engine Title | It is used for SEO purpose. | |
| SEDescription | Text | Search Engine Description | It is used for SEO purpose. | |
| SEKeywords | Text | Search Engine | It is used for SEO purpose. | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

| | | | | |
|---|---|---|---|---|
| | | Keywords | | |
| SEAltText | Text | Search Alter Text | It is used for SEO purpose. | |
| OrderProductByLooks | Boolean | 0 or 1 | If 1 then most looks product displayed first. | |
| Address1 | Text | Manufacturer Address1 | It defines the Address of the Manufacturer. | |
| Address2 | Text | Manufacturer Address2 | It defines the Address of the Manufacturer. | |
| Suite | Text | Suite | It defines the Suite/ Apt.# of the Manufacturer. | |
| City | Text | City Name | It defines the City of the Manufacturer. | |
| State | Text | State Name | It defines the State of the Manufacturer. | |
| ZipCode | Numeric | ZipCode | It defines the zipcode of the Manufacturer. | |
| Country | Text | Country Name | It defines the country of the Manufacturer. | |
| Phone | Text | Phone Number of Manufacturer | It defines the phone number of the Manufacturer. | |
| Fax | Text | Fax | It defines the fax number of the Manufacturer. | |
| URL | Text | Web Url Manufaturer | It defines the WebUrl of the Manufacturer. | |
| Email | Text | Email Address of Manufaturer | It defines the Email Address of the Manufacturer. | |
| Published | Boolean | 0 or 1<br>If 0 then Manufacturer will not appeared when browsing | 0-No, 1-Yes. This option determines whether or not the manufacturer will appear in the menus/sitemap. | |
| Deleted | Boolean | 0 or 1 | 0 means record is not deleted, 1 means record is deleted. | |
| AllowDuplicate | Boolean | 0 or 1 | 0 means not allow creating a duplicate entry with category name. | |
| PowerSearchEnable | Boolean | 0 or 1 | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer and if the Value = 1, the Attribute Filter will be displayed in Manufacturer Page otherwise not. | |
| PowerSearchColumnWidth | Numeric | 4 | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer then you can specify the Number of Column per row to be displayed in the Manufacturer Grid View. | |
| PowerSearchSelectionType | Text | Power Search Selection Type. e.g. : Single | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer then you need to Specify the SelectionType = "Single" | |
| PowerSearchAttributeClass | Text | Power Search Attribute Class. e.g. : Attribute Class Name | If you are using Stock Version of ASPDNSF then ignore this Column. Else if you are a SEE Customer then you need to Specify the ProductClass Name which is belongs to the Attributes. | |
| ImageType | Text | Type of Image. e.g. : Icon;Medium;Large | Type of Image should be define with semicolon (;) separation. | |
| SourceImagePath | Text | Image Path, e.g. : C:\Image Path | It defines the valid path of the source images. | |
| ImageName | Text | Image Name e.g. : Image1.jpeg | It defines the valid name of the image. File extension should be either “.jpeg/.jpg” or ”.gif” or “.png” | |

Note: When Action is “Delete” the only required fields are ManufacturerID and the Action Column.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

## KIT CSV DATA AND COLUMN NAMES

The table below lists the allowed data and the type of data for each of the columns (properties) for Entities. The notes section explains any precautions or limitations associated with the particular property (column).

| Column Name | Type | Example | Description | Notes |
|---|---|---|---|---|
| ProductID | Numeric | 6 | Enter the ProductID for which you want to Insert the Kit Item. Here Kit should insert for Product whose id is 6. | |
| Group:Name | Text | KitGroupName e.g. Color | It defines the Kit Group Name. | |
| Group:Type | Text | KitGroupType e.g. Single Select Dropdown List | It should be either Single Select Dropdown List or Multi Select Checkbox or Radio button, etc. | |
| Item:Name | Text | Kit Item Name e.g. TextItem | It defines the name of the kit item. | |
| Action | Text | Insert/Update/Delete | If Insert: Kit Item should be inserted into the master table.<br><br>If Update: Kit Item should be updated according to value in ProductID, Group:Name and Group:Type columns.<br><br>If Delete: Kit Item should be deleted according to value in ProductID, Group:Name and Group:Type columns. | |
| SKU | Text | Enter the SKU for which you want to Insert the Kit Item. | It defines the SKU of Kit Item. | |
| Group:Description | Text/HTML |  Description of the Kit Group in HTML Content.  | It defines the description of the Kit Group either in HTML content or Simple Text. | |
| Group:DisplayOrder | Numeric | Kit Group DisplayOrder e.g. 1 | It defines the display order of Kit Group. | |
| Group:Required | Boolean | 0 or 1 | If 1 then Kit Group Selection required. | |
| Group:ExtensionData | Text/HTML | Simple Text Or HTML for Group Extension Data | It defines any value you want to display for group either simple text or HTML. | |
| Item:Description | Text/HTML | Test ItemDesc. | It defines the description of the Kit Item. | |
| Item:DisplayOrder | Numeric | 1 | It defines the order of kit item. | |
| Item:Default | Boolean | 0 or 1 | 1 means default kit item. It should be only one default kit item for each “Item:Name” | |
| Item:DeltaPrice | Numeric | 10.00 | It defines the Price of the Kit Item. | |
| Item:DeltaWeight | Numeric | 0.05 | It defines the weight of the Kit Item. | |
| Item:ExtensionData | Text/HTML |  Kit Item ExtensionData …  | It defines any value you want to display for Kit Item either simple text or HTML. | |
| Item:SourceImagePath | Text | C:\Path | Valid image path for Kit Item image. | |
| Item:ImageType | Text | Icon;medium;large | Type of Image should be define with semicolon (;) separation. | |
| Item:ImageName | Text | kit1.jpg | It defines the valid name of the image. File extension should be either “.jpeg/.jpg” or ”.gif” or “.png” | |

Copyright © 2003 - 2013 Simrun Technologies L.L.C.

# Chapter 7: ASPDNSF COMMANDER R3 LICENSE

## REQUESTING A NEW LICENSE

After you have purchased Aspdnsf Commander R3 from either the Aspdotnetstorefront Marketplace, from Simrun Authorized Consultant, or directly from Simrun you will need to complete a form to get your license, to do this, please go to the following web page:

http://www.simrun.com/AspDotNetStoreFront/AspdnsfCommander/ASPDNSFCOMMANDERLicense.aspx

Here please fill out your order information and choose the "Request a New License" in the License Request drop down option. Once you have submitted your request allow 1-2 business days to process your request. Once processed a license file will be emailed to you.

To install this license, simply place it inside the directory that you installed Aspdnsf Commander R3 and then restart it, if installed correctly, your license information will be displayed in the About tab.

## UPGRADE YOUR R2 LICENSE

If you have purchased R2 and are now looking to upgrade to R3, you can request an UPGRADE to R3, simply follow the same steps as you would if you are requesting a new license but choose Upgrade Existing License in the License Request drop down. You will need type in your OLD key into the Commander R3 Key and select whether the key is for Single Store or Multi Store.

Please note, customers who purchased Aspdnsf Commander R2 after September 1st 2011 are eligible for a free upgrade, customers who purchased Aspdnsf Commander R2 before this will need to purchase an upgrade at 50% off the list price (you may contact Simrun directly at info@simrun.com) to upgrade your old license or purchase this from the Marketplace.

Copyright © 2003 - 2013 Simrun Technologies L.L.C.